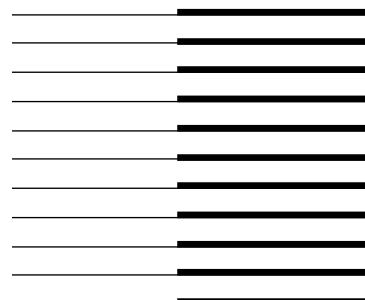




# Océ 9800

## RCF/Job Ticket Interface

*Programmer's Manual*





## Océ-Nederland B.V.

This manual documents the Release 2.0 software of the Océ 9800 controller. The manual (optionally available with the Océ 9800) is to be used by system programmers to prepare the host environment for the Océ 9800. Chapter 3 of this manual describes the job ticket data format of the Océ 9800 ReproStation release 2.0.

### Trademarks

HP-GL, HP-GL2 are trademarks of Hewlett-Packard Company

Calcomp is a trademark of California Computer Products, Inc.

PostScript is a trademark of Adobe Company

Centronics is a trademark of Centronics Corporation

### Copyright

Océ-Nederland B.V. Venlo, The Netherlands © 1996

All rights reserved. No part of this work may be reproduced, copied, adapted, or transmitted in any form or by any means without written permission from Océ.

Océ-Nederland B.V. makes no representation or warranties with respect to the contents hereof and specifically disclaims any implied warranties of merchantability or fitness for any particular purpose.

Further, Océ-Nederland B.V. reserves the right to revise this publication and to make changes from time to time in the content hereof without obligation to notify any person of such revision or changes.

---

# Contents

---

## Chapter 1

### **Introduction**

Concepts 10  
The Océ 9800 default mechanism 12

## Chapter 2

### **Remote Control Format (RCF)**

Job types 16  
RCF structure 17  
    APPLDATA structure (notation conventions) 18  
    APPLDATA command arguments (notation conventions) 18  
    APPLDATA comments (notation conventions) 19  
    Data flow example to the Océ 9800 20  
RCF behaviour 22  
    Decoding 22  
    Incomplete parameters set 22  
    Mandatory parameters 22  
APPLDATA commands 23  
Pen attributes (APPLDATA 001) 24  
    Syntax 24  
    Description 24  
    Examples 25  
    Behaviour 25  
    Patterns 25  
Job parameters (APPLDATA 002) 26  
    Syntax 26  
    Description 26  
    Single jobs 28  
    Set jobs 28  
    Matrix jobs 30  
Transformations (APPLDATA 003) 35  
    Syntax 35  
    Description 35  
    Examples 37  
    Behaviour 38  
Stamp (APPLDATA 020) 39  
    Syntax 39  
    Description 39

Examples	40
Media selection (APPLDATA 021)	41
Syntax	41
Description	41
Examples	43
Behaviour	43
Finishing (APPLDATA 022)	44
Syntax	44
Description	44
Example	45
Delivery (APPLDATA 023)	46
Syntax	46
Description	47
Examples	48
Edge correction (APPLDATA 024)	49
Syntax	49
Description	49
Examples	50
Behaviour	50
Framing (APPLDATA 025)	51
Syntax	51
Description	51
Examples	52
Behaviour	52
Original to copy matrix (APPLDATA 026)	53
Syntax	53
Description	53
Examples	54
Behaviour	54
Scan to file (APPLDATA 028)	55
Syntax	55
Description	56
Examples	57
Alignment (APPLDATA 029)	58
Syntax	58
Description	58
Example	59
HP-GL emulation (APPLDATA 050)	61
Syntax	61
Description	61
Example	62
HP-GL/2 emulation (APPLDATA 051)	63
Syntax	63
Description	63
Example	64

ASCII emulation (APPLDATA 056)	65
Syntax	65
Description	65
Example	66
CalComp emulation (APPLDATA 058)	67
Syntax	67
Description	67
Example	68
Automatic language sensing (APPLDATA 059)	69
Syntax	69
Description	69
Example	70
Scanner info/data (APPLDATA 106)	71
Syntax	71
Description	72
Example	73
Typical examples of RCF jobs	75
Single file	75
Sets	76
Matrices	78
APPLDATA Quick reference table	83

---

### Chapter 3

<b>Océ 9800 Repro Station</b>	
<b>Job Ticket Format specification</b>	
Introduction	90
General description	92
What is a job ?	92
What is a job ticket?	93
Token	94
Block structure	94
Output block	95
Definition block	96
Detailed description	100
Syntax	100
Token	102
Job ticket layout	103
Block structure	104
Input/output handling	104
Direct input	105
Indirect input	105
Order of processing	107

Job Ticket keywords 108  
Structure keywords 109  
    BeginBlock 109  
    BeginOutput 109  
    BeginTicket 109  
    EndBlock 110  
    EndOutput 110  
    EndTicket 110  
    IncludeBlock 110  
Setting keywords 112  
    Account 112  
    AddStrip 112  
    ASCII 113  
    CalComp 113  
    Collate 114  
    Comment 115  
    Copies 115  
    CopyMatrix 115  
    Copyright 117  
    CreationAppl 117  
    CutMethod 117  
    Directory 118  
    Distribution 118  
    Emulation 118  
    Fold 119  
    HPGL 119  
    HPGL/2 120  
    JobCollate 120  
    JobFlagsheet 120  
    JobName 121  
    Matrix 121  
    MediaFeed 121  
    MediaType 122  
    Mirror 122  
    Name 123  
    Notes 123  
    OrigDirectory 123  
    OrigName 124  
    OutputSize 124  
    Pens 124  
    Punch 125  
    RemoveStrip 125  
    Rotate 125  
    Shift 126  
    Stamp 126

UserName	127
Zoom	127
Overview	129
Implementation notes	131
Handling job tickets	131
Typical examples for Océ 9800 Repro Station	133
Example 1 Normal job	133
Example 2 Matrix job	136

---

## Appendix A

<b>List of error codes</b>	
Set memory errors	142
Océ 9800 controller error codes	144

---

## Appendix B

<b>Data formats</b>	
HP-GL commands	148
HP-GL escape sequences	151
HP-GL/2 commands	152
HP-RTL commands	154
CalComp commands	156
Calcomp command overview	156
Command details	160
Error conditions	163
Interface requirements	163
ASCII emulation	164
TIFF	166
CALS	168

---

## Appendix C

<b>RCF grammar</b>	
RCF grammar	170

---

## Appendix D

<b>Miscellaneous</b>	
Notation conventions	176
Reader's comment sheet	177
Related documentation	179
Index	181



---

Océ 9800 RCF/Job Ticket

---

*Programmer's Manual*

---

---

---

## Chapter 1

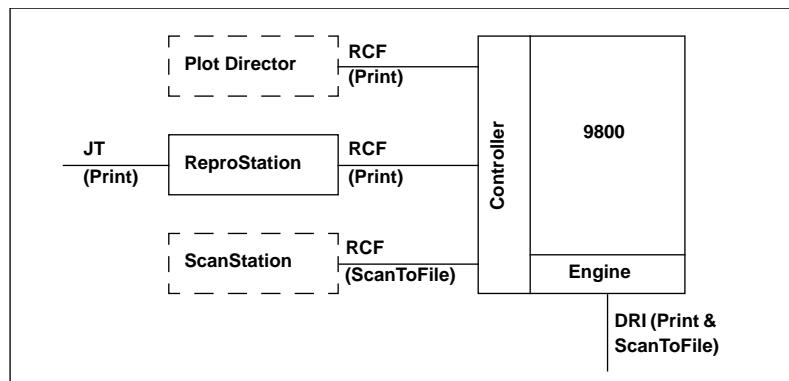
### Introduction

*Data streams at different interface levels, and a description  
of the default mechanism.*



# Concepts

The following data streams can be used at different interface levels:



[1] Océ 9800 data streams at different interface levels

**Job Ticket (Print)** The JT format is used by the Windows program Job Director, to send plot files to the Repro Station. One or more files are collected in a Job Ticket. The software allows users to specify per file or per group of files, file parameters and almost every function of the print engine. Next to this, instructions for the print room operator can be attached to a job ticket. Repro Station runs on a PC located close to the Océ 9800 print engine. The print room operator uses Repro Station to view and manage incoming print jobs. Jobs can be modified, or cancelled.

The Job Ticket format specification explains format and syntax of the Océ job ticket language as defined for the Océ 9800 Repro Station. It is contained in chapter 3, 'Océ 9800 Repro Station Job Ticket Format specification' on page 89 of this manual.

**Remote Control Format (Print)** The RCF (Print) format is used by print applications like Plot Director and Repro Station, to send control information and plot data to the Océ 9800 controller. The control information is contained in Remote Configuration Headers (RCHs) and in Matrix Programs (MPs). RCHs are located in front of the plot files, MPs are located at the beginning of matrix jobs.

Each time a plot file is sent to the printer, it is preceded by an RCH which holds the information about the plot parameters for the Océ 9800 printer. Plot files

without accompanying RCH use the default settings as defined by the key operator of the Océ 9800.

**Remote Control Format (ScanToFile)** The RCF (ScanToFile) format is used by Scan Station for interaction with the Océ 9800. The RCF (ScanToFile) *syntax* is described in this manual. For information about the *protocol*, contact your Océ representative.

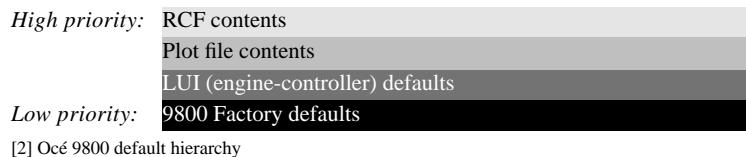
**Direct Raster Interface (Print & ScanToFile)** The DRI format can be used to send data directly to the Océ 9800 engine. In this way, you can develop your own controller, and connect it to the Océ 9800. The DRI format is described in the Direct Raster Interface Programmer's Manual.

## The Océ 9800 default mechanism

To understand the way the printer obeys commands, you need to know which commands are more important than other commands. In other words, which command level prevails over the other command level.

In the Océ 9800 four command levels are distinguished:

- commands in RCF files
- commands in plot files
- settings done on the Océ 9800 operating panel by the Key Operator
- Océ 9800 factory defaults



Settings can be changed at several levels. The lowest level is formed by the *factory defaults*, as described in the Océ 9800 User Manual.

On top of the factory default layer is the *LUI* (Local User Interface) *setting* layer. That is the site-specific definition of default settings by the Océ service engineer or the key operator.

On top of the LUI default layer is the *plot file* contents *setting* layer. The values of this layer are determined by the application that generates the plot file. The plot file may include format-specific envelopes (e.g. TIFF).

The top layer defaults are formed by RCF settings. This level of settings is the most important of all. Settings that have been done in this level will overrule the settings in lower layers. Note that RCF headers are not mandatory; they can or cannot be sent with the plot file.

If you use Job Tickets (JT) for Repro Station:

<i>High priority:</i>	JT settings
	RS defaults
	Plot file contents
	LUI (engine-controller) defaults
<i>Low priority:</i>	9800 Factory defaults

[3] Océ 9800 default hierarchy



---

Océ 9800 RCF/Job Ticket

---

*Programmer's Manual*

---

---

---

---

## Chapter 2

## Remote Control Format (RCF)

*Description of RCF, including APPLDATA commands in  
numeric order.*



---

## Job types

The Océ 9800 recognises three types of jobs:

- single jobs
- set jobs
- matrix jobs.

**Single jobs** contain only one plot file. They are preceded by an RCH that specifies the job as ‘single’ or they lack the RCH, in which case they use the settings pre-defined by the key operator.

**Set jobs** consist of a number of plot files. All plot files in a set can be processed together. The parameters applied to the set automatically apply to all files in the set. In this way it is possible to make several copies of sets of plot files, while you only specify the parameters once. The first plot in the set is preceded by an RCH that marks the beginning of the set and defines the default for all files in the set. Each individual plot file in the set may be preceded by an RCH that applies to the following plot only (overruling set parameters.)

**Matrix jobs** are advanced set jobs. Different functions can be performed on the same set of plot files. This allows the same set of files to be printed in various settings, such as the combination of plots, the scale factor, the media on which to be printed, etc. This way, users can create a number of different print jobs based on one set.

Matrix jobs consist of one or more matrix programs, RCHs and plot files. RCHs can be recognised by the ‘Type=Header’ parameter in the MFDESC string. Each matrix program can be recognized by the ‘Type=MatrixPrg’ parameter in the MFDESC string. All of this is explained in the next section called ‘RCF structure’ on page 17.

---

## RCF structure

The basic structure of the Oc  Remote Control Format (RCF) adheres to the CGM specification:

```
BEGMF "<title>" ;
MFVERSION 1;
MFDESC "Oce RCF, Version=2.0, Type=<type>, Unit=<unit>" ;
<Oce Application data>
ENDMF;
```

The keywords BEGMF, MFVERSION, MFDESC, and ENDMF are required in any CGM, and have the following meaning:

**BEGMF** flags the start of the CGM file. It accepts a quoted character string parameter, which allows the creator (the user or the application) to add an identification to the file. The length of the string should be less than 127 characters.

**';** the semi-colon is used as command delimiter.

**MFVERSION** is a required CGM command, specifying the CGM version (currently 1).

**MFDESC** takes a string parameter, which will be used by Oc  to identify the version number of our remote control format, as well as the type of RCH. This string is case-insensitive.

**Version** must be "2.0" for RCHs conforming to RCF version 2.

**Type** is either "Header", "MatrixPrg" or "ScanToFile".  
*This string is case-sensitive!*

**Unit** is either "Metric" or "Inches". This defines how things like pen widths, margins, etc. are to be interpreted. The Unit keyword is not supported.

**<Oce application data>** a number of lines containing the parameters, see next section.

**ENDMF;** specifies the end of the CGM data. Note the semicolon! Note that any following line break is part of the plot data, although examples may give the (false) impression that it still belongs to the header. This is especially important when the following plot is an ASCII file.

---

## APPLDATA structure (notation conventions)

The CGM Clear Text Application Data format is defined as any number of lines conforming to the following syntax:

```
APPLDATA <command number> <data record> <terminator>
```

These fields must be separated from each other by one or more of the following characters:

space, hor-tab, vert-tab, carriage-return, line-feed, form-feed.

**APPLDATA** requires an integer and a string as its arguments.

**command number** is an ASCII digit string, representing an integer number which identifies the command. It is limited to 3 digits, and "003", "03", and "3" are all equivalent.

**data record** (also known as command argument) This is a single- or double-quoted string, containing a list of parameters for the requested command (see below).

**terminator** The terminator separates the different apldata elements. It is either a semi-colon (";") or a slash ("/").

---

## APPLDATA command arguments (notation conventions)

As mentioned above, the command argument (data-record) is a (either single- or double-) quoted string which specifies the parameters. It contains a (possibly zero-length) sequence of "key=value" pairs, separated by a comma. Whitespace (as defined above) is allowed both before and after the comma, the string is limited to 3000 bytes.

The key is a (case-insensitive) string of exactly two characters of which the first must be alphabetic, and the second alphanumeric (i.e. [a-zA-Z][a-zA-Z0-9]). The value may be an integer, a fixed-point number, a string, a list, or a range:

**Integer(num)** This is an positive whole number in the range [0..999999999] (that are nine 9's). This is the maximum range for the decoder, every command has its proper limits. Note that you must specify an integer when so requested, "1.0" instead of "1" results in an error!

**Fixed-point(fxp)** This is a positive number consisting of maximal 4 significant digits before, and max. 4 digits after the decimal point (so the allowed range is [0000.0000 .. 9999.9999]). Note that "1", "1.", "1.0", and "001.00" all indicate the same fixed point number "0001.0000".

**String(str)** is limited by the following comma or the trailing quote of the data record. This means that a string cannot contain these characters! When specifying a string, don't make it longer than specified for the parameter in question.

**List(lst)** is a list of non-negative numbers, enclosed by a '<' '>' pair. The numbers are not necessarily ordered in any way, and may be separated from each other by a space or a comma. So "<1 3 2 34>" is the same as "<34,3,2,1>". Double entries are ignored.

**range** This is a pair of two non-negative numbers, separated by a "-". The first value is the start value. The last value is the end value. The first value is smaller than the last value.

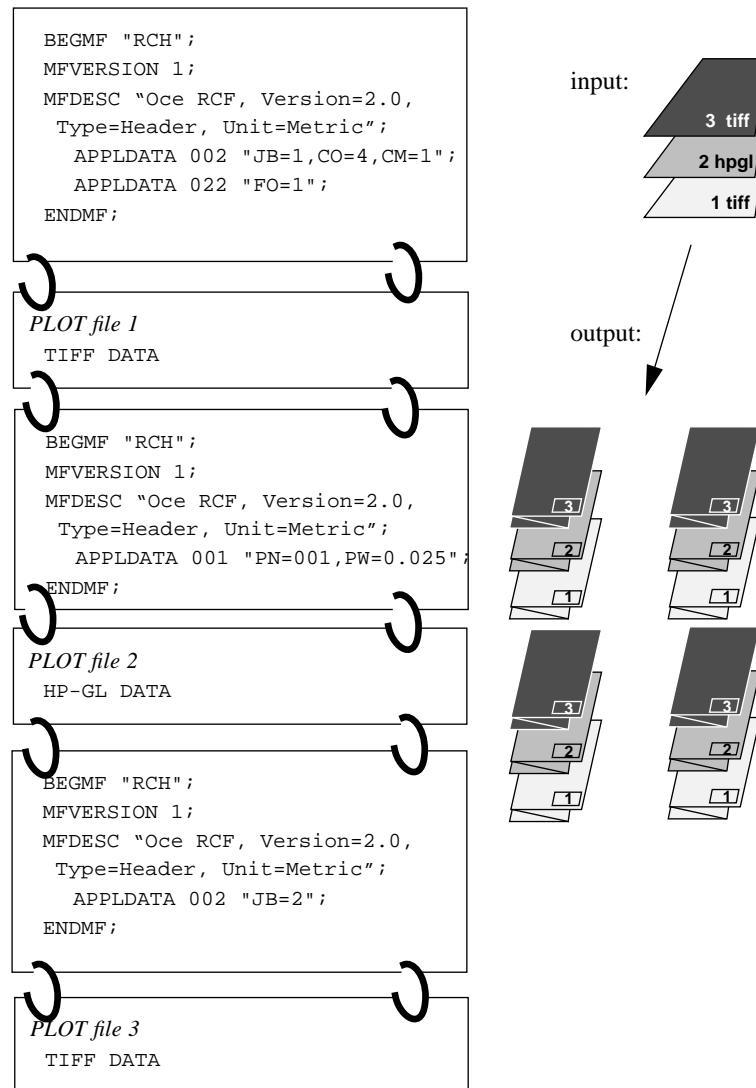
---

## APPLDATA comments (notation conventions)

Comments in the metafile must be embedded between "%" characters, everything between a pair of "%" -s is skipped.

Note that this comment format belongs to the metafile, it should not be used outside the BEGMF ... ENDMF.

## Data flow example to the Océ 9800



[4] Set job example, 4 sets of three files folded.

The example data flow in illustration 4 on page 20 shows you how three plot files need three RCHs to be printed as a set.

The first RCH marks the start of the job (APPLDATA 2 "JB=1") and defines the job settings. It defines how much copies (APPLDATA 2 "CO=4") are needed and that the output must be folded (APPLDATA 22 "FO=1").

The third RCH marks the third plot file and defines the end of the job (APPLDATA 2 "JB=2").

RCHs define job boundaries and print settings for the job.

**Note:** *The Oc  9800 delivers the pages face-up. That is why file 3 is on top.*

# RCF behaviour

## Decoding

The RCF decoder starts by reading the data. The decoder expects the format of the data to be as described in the previous section, and it verifies the version in the MFDESC argument string.

The APPLDATA lines will be processed one by one. They will be checked on syntax before they will be applied. If the same parameter will be encountered more than once the last occurrence will be used.

The RCF processing will end when the ENDMF; keyword has been encountered.

**Note:** *Océ service may have configured the machine to work with inches or with the metric system. Bear in mind that the remote control values must be sent in the units the machine is currently using. All examples in this document are in units of the metric system, including ranges.*

**Note:** *Multiple RCHs before a plot will be concatenated. That is, they will be considered as one. When parameters have been defined twice or more, the last definition is valid.*

## Incomplete parameters set

If not all possible parameters of an APPLDATA command are specified, the attributes that are not explicitly specified will be left unchanged.

For example, a pen width can be changed without specifying a pen pattern, which means that the pattern of the default pen, as defined by the Océ 9800 Key Operator will be used (see page 24).

## Mandatory parameters

A command can also have so-called mandatory parameters, such as the PN (Pen Number) parameter of APPLDATA 1: Pen Attributes. This parameter is mandatory because you can't define a pen width without specifying the pen itself.

---

## APPLDATA commands

The data stream to the Océ 9800 controller consists of two different sets of files: a sequence of plot files which contain the information to be printed and control data which contain instructions on how to perform the action.

```
BEGMF "RCH";
MFVERSION 1;
MFDESC "Océ RCF, Version=2.0,
Type=Header, Unit=Metric";
APPLDATA 002 "JB=0,CM=1,CO=4";
APPLDATA 022 "FO=1";
ENDMF;
```

*PLOT file*  
TIFF

[5] Example of a simple job consisting of two parts: an RCH and a plot file

This rest of this chapter is about the syntax and meaning of the APPLDATA command line as will be used in the RCF to control the Océ 9800 print process.

# Pen attributes (APPLDATA 001)

## Syntax

**APPLDATA 001** "PN=PenNumber, PW=PenWidth, PP=PenPattern";

parameter	meaning	type	default
<i>PenNumber</i>	a single pen specified by a pen number, 1..999 range of pens: n1-n2, 1 <= n1 <= n2 <= 999	<num> <n1>-<n2>	
<i>PenWidth</i>	pen width in millimetres or inches: 0.12mm or 0.025" -16.25 mm/0.639" at 400 dpi	<fxp>	0.19 mm 0.0079"
<i>PenPattern</i>	pen pattern: 1..16 are grey scales, 17..32 are Océ patterns	<num>	16 (black)

[6] Pen attributes

## Description

The *Pen attributes* command is used to define pens, width, pattern.

**PN Pen Number** is a mandatory parameter specifying either a single pen or a range of pens to be modified. Note that you cannot modify pen 0, although it is a pen like all others in HP-GL/2. The range is specified using a hyphen ‘-’: 2-45 is pen two to forty-five.

**PW Pen Width** is an optional parameter specifying the width of the pen(s), which will be rounded to the nearest supported number of pixels.

The supported range is machine-dependent. The pen width may be set to 0.12 mm, which will result in a line with the smallest width (for example a two pixel line on the 9800-series machines.)

**PP Pen Pattern** is an optional parameter. 32 patterns are available of which the first 16 (1-16) correspond to increasing shades of gray (1=white, 16=black), the next 16 (i.e. 17-32) are Océ defined patterns.

---

## Examples

Define pens 1 to 10 as transparent grey, width 1mm.:

```
APPLDATA 01 "PN=1-10, PW=1.0, PP=4";
```

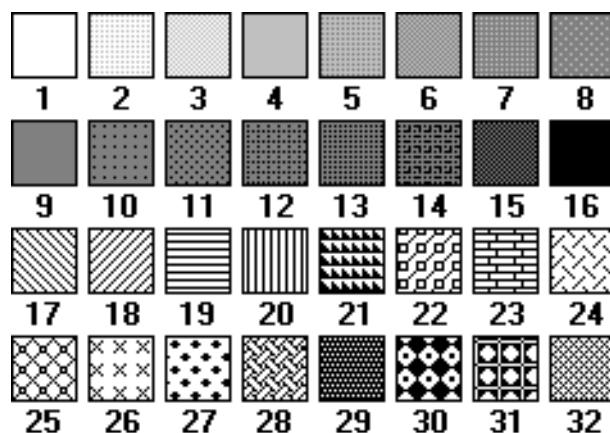
---

## Behaviour

- When the same pen is redefined multiple times, the last definition applies.
- Only the pen number parameter is mandatory. The others may be omitted, leaving the corresponding pen attributes unchanged.
- One APPLDATA must be sent for each different pen configuration.

---

## Patterns



---

## Job parameters (APPLDATA 002)

---

### Syntax

**APPLDATA 002** "CO=COpies, AI=AccountId, UI=UserId, JB=JobBoundary, CM=CopyMethod, PL=PlotList, PN=PlotNumber";

parameter	meaning	type	default
<i>COpies</i>	number of copies to print, 1..999	<num>	1
<i>AccountId</i>	account number (max 9 digits)	<num>	0
<i>UserId</i>	user number (max 9 digits)	<num>	0
<i>JobBoundary</i>	SINGLE FILE(0), START OF SET(1) or END OF SET(2)	<num>	0
<i>CopyMethod</i>	Sort copies: BY PAGE(0) ( <i>Only if JB=0!</i> ) or BY SET(1)	<num>	0
<i>PlotList</i>	list of plots for the specified matrix program	<lst>	0
<i>PlotNumber</i>	number of the plot inside the matrix job	<num>	0

[7] APPLDATA 002 job parameters

---

### Description

This command defines the parameters of the different jobs that may be used: single plot file jobs, sets of file jobs or matrix jobs. The parameters of this APPLDATA may be found in the plot header (RCH) or in the matrix program (MP).

Files in a set belong together. When printed, the order will be kept intact. Three copies of a set that consists of file1, file2 and file3 result in the following output: 123,123,123.

**CO** *Number of copies* to plot.

**AI** *Account Id* Nine digit number that is used for accounting to identify the account name of the submitter of the job.

**UI** *UserId* Nine digit number that is used for accounting to identify the account name of the submitter of the job.

**JB** The *JobBoundary* parameter defines the type of job: a single job or a set job.

Single job: the job consists of only one plot, the *JobBoundary* parameter is not needed (it defaults to 0, single file).

Set job: the data stream contains two APPLDATA 002 command lines. The first contains a JB=1 (start of set) marker. The second APPLDATA 002, which is placed before the last plot file, has a JB=2 marker (end of set).

**CM** The *copy method* parameter specifies the way copies should be sorted: by page (0) or by set (1).

**Attention:** *Copy by page (CM=0) may only be selected if the JobBoundary is set to Single File (JB=0).*

**PL** The *plot list* parameter specifies the plot files of the matrix job. Specify all plot numbers (see PN parameter).

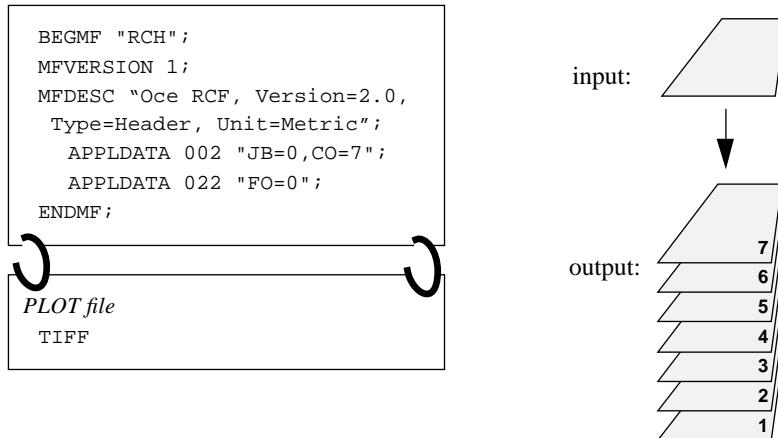
**PN** The *plot number* parameter specifies the sequence *number* of the plot in the matrix job. The matrix program(s) use it to identify the plot.

The following table shows where the parameters of this APPLDATA must be used in function of the type of job. RCH is the remote configuration header. SH is the set header, the RCH that contains the set start. MP is the Matrix Program.

Parameter	<i>jobtype</i>		
	single	set	matrix
<i>Copies</i>	RCH	SH	MP
<i>CopyMethod</i>	RCH	SH	MP
<i>PlotList</i>	-	-	MP
<i>PlotNumber</i>	-	-	RCH

## Single jobs

Single jobs consist of two files, the RCH header and the plot file. If the RCH is omitted, the controller defaults are used.



[8] Single plot file job example, 7 copies of a tiff file unfolded

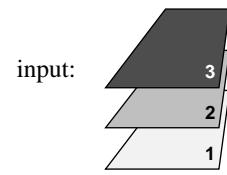
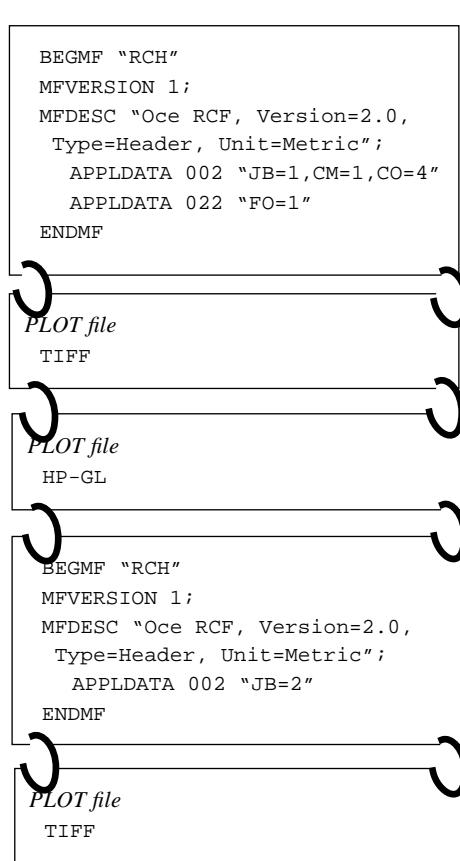
## Set jobs

A set job consists of at least two RCHs and a sequence of plot files. The first RCH is situated before the first plot file, the last RCH before the last plot file. The first RCH contains a set start (JB=1) command, the last RCH contains a set end (JB=2). If the set end is not present, the next job will be considered part of the set.

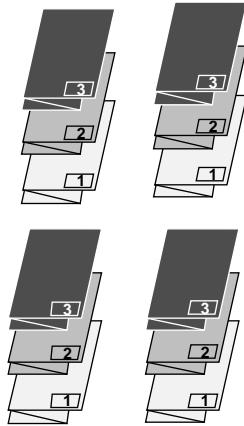
The first RCH defines the defaults for the whole set. For parameters that are not specified in the first RCH, default controller values are used. Additional RCHs, if any, are located in front of the plot file of the set, they apply to the following plot only.

**Attention:** *Copy by page (CM=0) may only be selected if the JobBoundary is set to Single File (JB=0).*

The example in illustration 9 on page 29 consists of a set of three plot files.



output:



[9] Set job example, 4 sets of three files folded.

## Matrix jobs

In matrix jobs, different functions will be performed on the same set of plot files, which is stored in the set memory. These functions are specified in matrix programs.

Each matrix program can be recognised by the ‘type=MatrixPrg’ parameter located in the MFDESC line of the RCH header. The start of the matrix job is marked by the first matrix program.

The RCH in front of the first plot after the matrix program(s) contains a set start, which is optional, because the first RCH that follows the last matrix program implicitly announces the first plot. The RCH in front of the last file contains a set end to indicate the end of the matrix job.

Each matrix program contains a list, specifying all plots of the set to be printed. The plots in the set are numbered with a parameter in the RCH. The numbering must be sequential (1, 2, 3, ...).

The first RCH defines the defaults for all the plots in the set. Possible RCHs preceding other plot files apply to the following plot only.

The table below shows whether the specified APPLDATA must be placed in the matrix program or in the RCH. Bear in mind that it can never be defined in both (except for APPLDATA 002 Job Parameters). If job parameters have been defined both in the RCH and in the MP, a job description error will be generated. In this case, the APPLDATA command will be ignored in the MP program, according to the table 10 on page 31.

If more than 40 matrix programs are defined, the additional matrix programs will be ignored. In this case a job description error will be generated.

The plot list contains a complete list of all plot numbers. Each number must increase with 1 for each program: 0,1,2,3 ... and not 2,3,1,0. If this order is not kept or one of the plot numbers is missing, the reference program will be ignored.

If a plot has no matrix program, it will use the previous one. In this case, a job description error will be generated.

APPLDATA	MP (Matrix Program)	RCH (Header)
01 Pen attributes	No	Yes
02 Job parameters	Partially	Partially
Required:	CO (Copies)) AI (Account Id) UI (User Id) CM (Copy Method) PL (Plot List)	JB (Job Boundary) PN (Plot Number)
	<i>Not used/allowed in matrix:</i> LP (Long Plot) JB (Job Boundary) ignored	
03 Transformations	Partially	Partially
Required:	XS (XScale factor 25-100%) YS YScale factor 25-100% MR (MiRrror) AS (AutoScale)	RO (ROtation) AR (Auto Rotate) LC (Legend Correction))
	<i>Not used/allowed in matrix:</i> ES (Enhanced Scale) SU (Shift Up) SD (Shift Down) SL (Shift Left) SR (Shift Right)	
20 Stamp	Yes	No
21 Media Selection	Yes	No
22 Finishing	Yes	No
23 Delivery	Yes	No
24 Edge correction	Yes	No
	<i>Not used/allowed in matrix:</i> LR (Leading edge Remove) TR (Trailing edge Remove)	
25 Framing	Yes	No
26 Original to Copy Matrix	Yes	No
29 Image alignment	No	Yes
50-59 Languages	No	Yes

[10] APPLDATA commands belonging in the MP and in the RCH

**Note:** If Job Boundary appears in the MP, it will be ignored. No job description error will be generated.

Bypass (manual feed) cannot be used in matrix jobs, if present it will be ignored. A job description error will be generated.

There can be up to 40 different reductions (scale factors), one for each matrix program. The controller will calculate the zoom ratio for each matrix program (auto zoom/size/rotate) before sending it to the engine.

If there are different print orientations (caused by the (auto) rotation setting) for one plot in different matrix programs, the controller reports a job description error and generates the bitmap according to the settings in the first referenced matrix program.

The matrix job example in illustration 11 on page 33 consists of 2 matrix programs and 3 plot files.

```

BEGMF "matrix program number 1";
MFVERSION 1;
MFDESC "Oce RCF, Version=2.0,
Type=MatrixPrg, Unit=Metric";
APPLDATA 002 "PL=<1,2,3>,CO=2, CM=0";
APPLDATA 022 "FO=1";
ENDMF;

BEGMF "matrix program number 2";
MFVERSION 1;
MFDESC "Oce RCF, Version=2.0,
Type=MatrixPrg, Unit=Metric";
APPLDATA 002 "PL=<1,2,3>,CO=4,CM=1";
ENDMF;

BEGMF "RCH number 1"; MFVERSION 1;
MFDESC "... Type=Header...";
APPLDATA 002 "JB=1,PN=1";
ENDMF;

PLOT file 1
CALS

BEGMF "RCH number 2";MFVERSION 1;
MFDESC "... Type=Header...";
APPLDATA 002 "PN=2";
ENDMF;

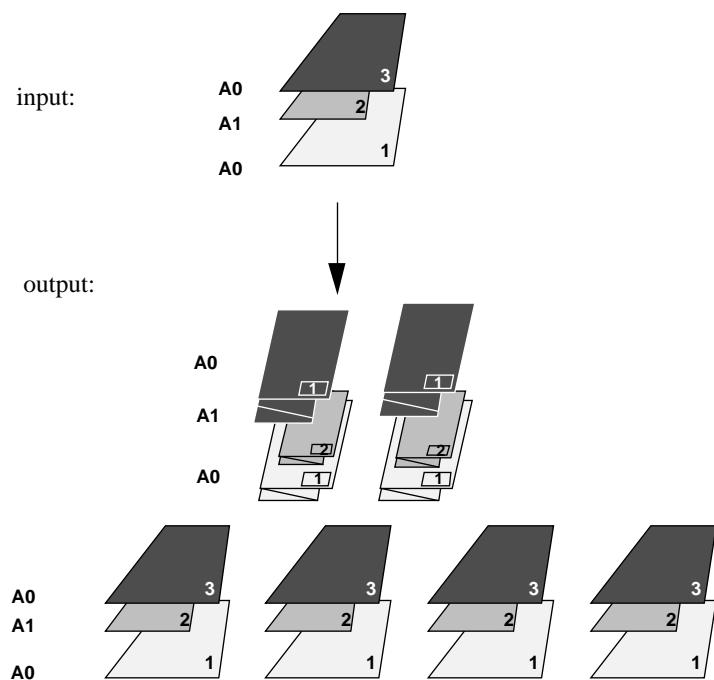
PLOT file 2
TIFF

BEGMF "RCH number 3";MFVERSION 1;
MFDESC "... Type=Header...";
APPLDATA 002 "JB=2, PN=3";
ENDMF;

PLOT file 3
HP-GL

```

[11] Matrix job example (part 1)



[12] Matrix job (part 2)

# Transformations (APPLDATA 003)

## Syntax

**APPLDATA 003** “XS=XScale, YS=YScale, RO=ROtation, MR=MiRrror,  
AS=AutoScale, AR=AutoRotate, LC=LegendControl, SU=ShiftUp,  
SD=ShiftDown, SL=ShiftLeft, SR=ShiftRight”;

parameter	meaning	type	default
<i>XScale</i>	X scale (or zoom) factor 0.2500..4.0000	<fxp>	1
<i>YScale</i>	Y scale (or zoom) factor 0.2500..4.0000	<fxp>	1
<i>ROtation</i>	rotation angle: 0, 90,180 or 270 degrees.	<num>	0
<i>MiRrror</i>	Drawing can be mirrored in the X direction: NO MIRROR (0) or MIRROR IN X (1)	<num>	0
<i>AutoScale</i>	automatic scale or zoom, DISABLED(0) or ENABLED(1)	<num>	0
<i>AutoRotate</i>	automatic rotation, DISABLED(0), FOLDING (1) or PRODUCTIVE(2)	<num>	0
<i>LegendCon-trol</i>	legenda control, DISABLED(0), or ENABLED(1)	<num>	0
<i>ShiftUp</i>	shift up factor: 0 to 1219mm/48inch	<fxp>	0
<i>ShiftDown</i>	shift down factor: 0 to 1219mm/ 48inch	<fxp>	0
<i>ShiftLeft</i>	shift left factor: 0 to 914mm/36inch	<fxp>	0
<i>ShiftRight</i>	shift right factor: 0 to 914mm/ 36inch	<fxp>	0

[13] Tranformation(ID=003) syntax

## Description

This command is used to *rotate, scale, or shift* the plot.

**XS** This is the scale factor along the X axis,  $0.2500 \leq \text{scale\_factor} \leq 4.0000$ .

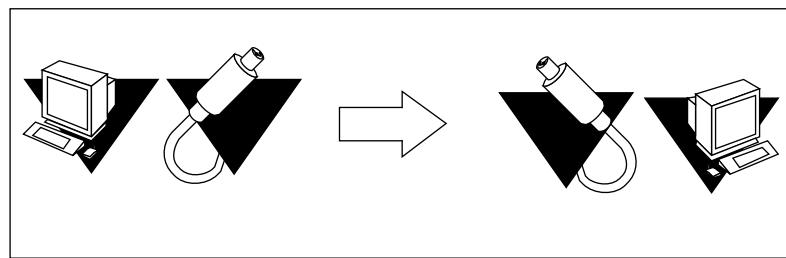
**YS** This is the scale factor along the Y axis,  $0.2500 \leq \text{scale\_factor} \leq 4.0000$ .

**Note:** *X and Y correspond to the paper axis, not to the axis of the drawing.*

**RO** The angle over which the drawing must be rotated (counter-clockwise), either no rotation (0 degrees), 90, 180, or 270 degrees.

**MR** indicates if the drawing must be mirrored in X direction:

- NO MIRROR (0)
- MIRROR IN X (1)



[14] Example of a drawing, mirrored in X direction

**AS** *AutoScale* (or *AutoZoom*) selects an automatic zoom, in function of paper format and plot size. If *AutoScale* is enabled, the *XS* and *YS* parameters are ignored.

**AR** *Automatic rotation* setting. This allows the plot to be automatically rotated or not, according to its value. Settings can be as follows:

- **DISABLED(0):** drawings are output as they are given to the controller. A roll is automatically chosen, depending on the other (rotate/paperformat) settings.
- **FOLDING(1):** drawings are rotated to have the legend on top if folded (depends on folding method).
- **PRODUCTIVE(2):** drawings are oriented landscape as much as possible to get the highest productivity. If landscape format is not available, portrait format is chosen automatically (e.g. A1 on A1-roll instead of on A0-roll).

**Note:** *Automatic Rotation will be overruled by Rotation (RO).*

**LC** *LegendControl* setting. This option allows you to rotate the image 180 degrees.

When AR (auto rotate) is set to folding (1), the Océ 9800 controller will print all drawings in portrait to ensure that the legend area is located in the proper (lower left) corner of the original. If this is not the case and the legend is located in the upper right corner, switch on LC to rotate the image 180 degrees extra to the lower left corner.

**Note:** The LC setting will be active for all possible values of the AR parameter. If AR has been disabled(0), LC cannot be set.

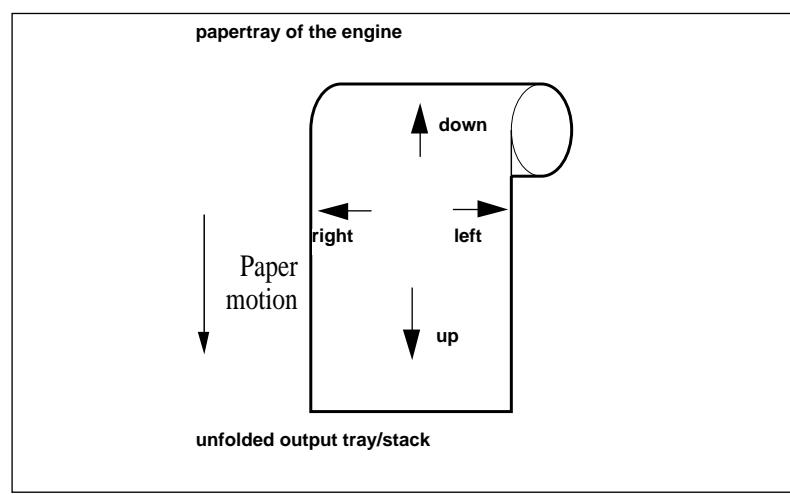
- ENABLED(1): image is rotated by 180 degrees.
- DISABLED(0): image is not rotated.

**SU** This is the *shift up* value along the paper feed axis.

**SD** This is the *shift down* value along the paper motion axis.

**SL** This is the *shift left* value along the cross-feed axis.

**SR** this is the *shift right* value along the cross-feed axis.



## Examples

Scale the plot by a factor 2 in both X and Y, rotate it 90 degrees:

APPLDATA 03 "XS=2, YS=2, RO=90";

---

## Behaviour

- *Shifts, AutoScale, FraMing, and Image Alignment* (APPLDATA 25) can only be used in an exclusive manner. If they are defined simultaneously, the last parameter is applied and a job description error is reported.

# Stamp (APPLDATA 020)

## Syntax

**APPLDATA 020** “SS=StampSelection, SN=StringNumber, PO=POsition, FS=FontSize, GS=GreyScale“;

parameter	meaning	type	default
<i>StampSelection</i>	stamping ENABLED(1) or DISABLED(0)	<num>	0
<i>StringNumber</i>	predefined string number, from 1 to 50	<num>	1
<i>POsition</i>	predefined stamp position: BOTTOM(0), MIDDLE(1) or TOP(2)	<num>	1
<i>FontSize</i>	predefined font: SMALL (0) or LARGE (1)	<num>	0
<i>GreyScale</i>	predefined grey scale choice: 0, 1, 2, 3 black = 0, darkgrey = 1, grey = 2, lightgrey = 3	<num>	0

[16] Stamp (ID=020) syntax

## Description

This command defines if, where and which *stamp* must be placed on the plot. The *stamp* number is defined from a list of predefined stamps, which have been defined using the Local User Interface (LUI).

**SS** This parameter indicates whether *stamp* is selected (1) or not (0).

**SN** This is the *string number* selected as stamp. 50 stamps are available, they correspond to predefined strings set on the Local User Interface of the Océ 9800.

**PO** *Stamp Position* selected for the stamp in the plot. Three predefined positions can be selected: at the BOTTOM(0), MIDDLE(1) or TOP(2) of the legend.

**Note:** With a long plot, the stamp will appear on the first A0/E+ page boundary.

**FS** Stamp *font size*, either SMALL(0) or LARGE(1).

**GS** Stamp *grey scale*, either BLACK(0), DARKGREY(1), GREY(2) or LIGHTGREY(3).

---

## Examples

APPLDATA 020 "SS=1, SN=10, PO=1, FS=1, GS=3";

This command selects the string number 10 to be added as stamp to the next plot in the middle position 1, with a large font size of 1 and a grey scale of 3 (LIGHTGREY).

# Media selection (APPLDATA 021)

## Syntax

**APPLDATA 021** “PF=PaperFormat, ME=MEdia, AF=AutoFormat, BY=BYpass, RM=RolloverrulingMethod”;

parameter	meaning	type	default
<i>PaperFormat</i>	format of the paper: A0, A1 ... (from 0/A0 to 21/700_MM)	<num>	AutoFormatresult
<i>MEdia</i>	medium choice: PAPER(0), TRANSPARENT(1) or POLYESTER(2)	<num>	0
<i>AutoFormat</i>	automatic format ENABLED(1) or DISABLED(0)	<num>	0
<i>BYpass</i>	manual feed ENABLED(1) or DISABLED(0)	<num>	0
<i>RolloverrulingMethod</i>	exactly the FIT (0), NEXT SIZE(1) or REDUCED(2)	<num>	0

[17] Media selection (ID=21) syntax

## Description

This command defines the format and type of the media on which the plot will be printed.

**PF** This is the *paper format* selection which specifies the roll width:

A0 = 0  
A1 = 1  
A2 = 2  
A3 = 3  
A4 = 4  
34" (E) = 5  
22" (D) = 6  
17" (C) = 7

11" (B) = 8  
8.5" (A) = 9  
36" (E+) = 10  
24" (D+) = 11  
18" (C+) = 12  
12" (B+) = 13  
9" (A+) = 14  
30" = 19  
500 MM = 20  
700 MM = 21

**ME** This is the *media* selection:

- PAPER(0)
- TRANSPARENT(1)
- POLYESTER(2)

**AF** *AutoFormat* command to select a *PaperFormat* automatically, so there will be no loss of information:

- AutoFormat DISABLED(0)
- AutoFormat ENABLED(1)

**BY** *Bypass* means manual feed. The operator will be requested to mount the proper paper format. It is used with the *PaperFormat* setting.

- Bypass DISABLED(0)
- Bypass ENABLED(1)

**RM** The *roll overruling method* describes how the plot will fit on the available rolls. It is defined to prevent that the engine has to stop, asking for the specified paper format to be loaded (operator interaction required). The following possibilities are available:

- *Fit*: Use the specified paper format. If it is not present, wait for it to be mounted by the operator. The controller sends the plot to the engine which will notify the operator via the scanner console.  
**Note:** *Print files in the small ANSI range (8.5", 11", 17", 22", 34") may lead to selection of the large ANSI format (9", 12", 18", 24", 36"), when the required (small ANSI) roll is not loaded in the printer.*
- *Next size*: Use the specified format. If it is not present, use the next larger available size. The controller sends the plot to the engine and selects a roll

which is the next larger. If not available, it behaves as *exact fit* (i.e. wait for operator).

- *Reduced*: Use the specified format. If it is not present, use the next available larger size. If this also is not present, use the next available lower size. The controller sends the plot to the engine with a zoom factor and an alternative roll selection. If not available (e.g. unavailable medium), it behaves as *exact fit* (i.e. wait for operator).

**Note:** The term ‘reduced’ may be misleading! Read this term as ‘best fit’.

## Examples

APPLDATA 021 “ME=0, PF=1, AF=0, BY=1”;

This command specifies manual feed (BYPASS) of A1 format, media type: PAPER.

## Behaviour

- *PaperFormat* (PF) has a higher priority than *AutoFormat*
- This APPLDATA may override internal format settings of the plots.  
This behaviour is language-dependent.
- If both *AutoScale* and *AutoFormat* are defined, the *original to copy matrix* is used, see APPLDATA 26.

# Finishing (APPLDATA 022)

## Syntax

**APPLDATA 022** “FO=FoldOptions, ME=foldMEthod, LE=LEngth, WI=WIdth, BE=BindingEdge, FR=FoldoRientation”;

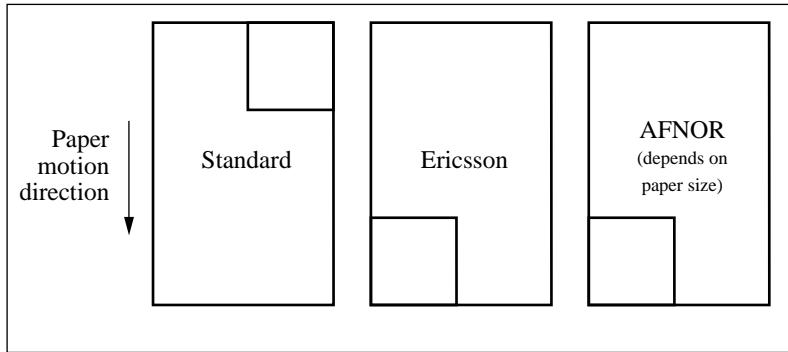
parameter	meaning	type	default
<i>FoldOptions</i>	FOLD OFF(0), FOLD ON(1) or FOLD AND PUNCH(2)	<num>	0
<i>foldMEthod</i>	STANDARD(0), ERICSSON(1) or AF-NOR(2)	<num>	0
<i>LEngth</i>	fold packet length from 276mm/10.9 inch to 310mm/12.2inch	<fxp>	297mm or 11"
<i>WIdth</i>	fold packet width from 186mm/7.3 inch to 230mm/9inch	<fxp>	210mm or 8.5"
<i>BindingEdge</i>	NONE(0) or a value in the range 15mm/0.6inch to 30mm/1.2inch	<fxp>	0
<i>FoldoRientation</i>	AUTO(0), or PORTRAIT(1)	<num>	0

## Description

This command activates folding, method of folding and punching.

**FO** Selection of *fold* and *punch options*: FOLD OFF(0), or FOLD ON(1) or FOLD AND PUNCH(2)

**ME** Selection of *folding methods*, depending on the position of the legend: STANDARD, ERICSSON or AFNOR. The STANDARD folding method is the same as the DIN, ANSI and architectural methods.



[18] Possible folding methods

**LE** The *fold packet length* must be between: 276 mm/10.9" and 310 mm/12.3"

**WI** The *fold packet width* must be between: 186 mm/7.3" and 230 mm/9"

**BE** The *binding edge* is the punch area. Select (0) when no binding edge should be added or select a value ranging from 15mm/0.59" to 30mm/1.18"

**FR** Selection of *fold orientation*: either in **PORTRAIT** format or **AUTO** format. Fold orientation is an engine setting. Use **AUTO** if you want to follow the folding method that belongs to the orientation of your file. Portrait files are folded differently than landscape file for each of three folding methods **DIN**, **Ericsson**, and **AFNOR**. Use **PORTRAIT** to force portrait folding method. For instance when your original contains a legend that requires portrait folding method.

## Example

APPLDATA 022 "FO=1, ME=1, LE=300, WI=200, BE=0";

This command line selects folding ON, with ERICSON folding method, packet length of 300mm, packet width of 200mm and no binding edge. Punch is not selected.

---

# Delivery (APPLDATA 023)

---

## Syntax

**APPLDATA 023** “CI=CutInfo, CL=CutLength, OU=OUtput, DE=DEposit,  
BN=BinNumber, JO=JOgging”

parameter	meaning	type	default
<i>CutInfo</i>	cut plot on the size of the PLOT(0) on the STANDARD SIZE(1), or on CUSTOM CUT (2)	<num>	0
<i>CutLength</i>	cut length 210 to 15000 mm 8.5 to 600 inch	<fxp>	210 mm 8.5 inch
<i>OUtput</i>	If Lower Container Unit installed: choosing upper or lower output to deliver plot: UPPER(0), LOWER(1) AUTO(2)	<num>	0
<i>DEposit</i>	If Folder with High Capacity Output Unit installed: deposit plot in the STACKER(0), on BELT1(1), on BELT2(2) or on EITHER BELT(3)  If High Capacity Stacker installed: CONTINUOUS (4) PER SET (5) PER JOB (6) BIN NUMBER (7)	<num>	0
<i>BinNumber</i>	If High Capacity Stacker installed, and BIN NUMBER is selected (DE=7): All copies are delivered onto the specified bin: 1..6	<num>	1
<i>JOgging</i>	If High Capacity Stacker installed: Jogging (offset stack) DISABLED(0) ENABLED(1)	<num>	0

[19] Delivery (ID=023) syntax

---

## Description

This command defines how the plotter handles finished plots: how they are cut and where they are delivered.

**CI** *Cut info* defines how plots are cut: on the size of the PLOT(0) (synchro) or on the length that belongs to STANDARD(1) format, or on length as defined in CUSTOM CUT (2).

**CL** *CutLength* determines the length of the plot when Custom Cut was selected. This parameter is ignored when CutInfo=0 (CI) is set to PLOT(0) or to STANDARD(1).

**OU** If Lower Container Unit installed, *Output* defines which output is used to deliver copies: the upper delivery output or the lower delivery output:

- **Upper output (0)**

The copies are delivered to a single receiving tray or the optional High Capacity Stacker.

- **Lower output (1)**

The copies can be collected under the lower delivery output, using an optional container for long copies.

- **Automatic (2)**

Normally the upper delivery output is used. Only if the copy is too long to be collected on the single receiving tray or High Capacity Stacker, the lower delivery output is used.

**DE** *Deposit* indicates where the output plot must be deposited:

If Folder is installed with High Capacity Output Unit:

- on the stacker (0), on belt1 (1), on belt2 (2), or on either belt (3).

If High Capacity Stacker is installed:

- **Continuous (4)**

The printer will switch to the next bin to deliver all copies. If this bin is full, the next bin will be used (bin 6 will be followed by bin 1). This process will continue until all bins are full. In that case printing is stopped.

- **Per Set (5)**

The printer will switch to the next bin to deliver all copies within the first set. Copies of the following set will be delivered at the next bin (bin 6 will be

followed by bin 1). This process will continue until the next bin is *completely* full (regardless the contents of other bins). In that case printing is stopped.

**Note:** If a set does not fit onto one bin, the next bin will be selected automatically to finish the set.

■ Per Job (6)

The printer will switch to the next bin to deliver all copies within the first job. Copies of the following job will be delivered at the next bin (bin 6 will be followed by bin 1). This process will continue until the next bin is *completely* full (regardless the contents of other bins). In that case printing is stopped.

**Note:** If a job does not fit onto one bin, the next bin will be selected automatically to finish the job.

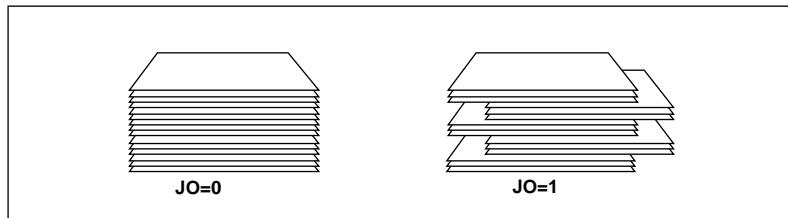
**Note:** One job may contain several sets, as defined in the RCF.

■ Bin number (7)

All copies are delivered onto the specified bin. This process will continue until the specified bin is full (regardless if another bin is empty). In that case printing is stopped.

**BN** If High Capacity Stacker is installed and Bin number is selected (DE=7), *BinNumber* specifies the number of the selected output bin.

**JO** If High Capacity Stacker is installed, *JOgging* indicates if jogging (offset stack) is DISABLED (0) or ENABLED (1).



[20] Jogging (Offset Stack)

---

## Examples

APPLDATA 023 "DE=0, CI=0";

This command line causes the plot to be delivered in the stacker and cut on the size of the plot.

# Edge correction (APPLDATA 024)

## Syntax

**APPLDATA 024** “LA= LeadingedgeAdd, LR=LeadingedgeRemove,  
TA=TrailingedgeAdd, TR=TrailingedgeRemove”

parameter	meaning	type	default
<i>LeadingedgeAdd</i>	0 to 297.4mm/11inch are added to the leading edge	<fxp>	0
<i>LeadingedgeRemove</i>	0 to 100mm/4inch are removed from the leading edge	<fxp>	0
<i>TrailingedgeAdd</i>	0 to 297.4mm/11inch are added to the trailing edge	<fxp>	0
<i>TrailingedgeRemove</i>	0 to 100mm/4inch are removed from the trailing edge	<fxp>	0

[21] Edge correction (ID=024) syntax

## Description

This command defines whether leading and/or trailing edge should be added or removed from the plot.

**LA** This setting adds a leading edge to the plot, expressed in mm or inches.  
This is done after the scale/zoom transform!

**LR** This setting removes a leading edge from the plot, expressed in mm or inches.  
This is done before the scale/zoom transform, after rotation.

**TA** This setting adds a trailing edge to the plot, expressed in mm or inches.  
This is done after the scale/zoom transform, after rotation.

**TR** This setting removes a trailing edge from the plot, expressed in mm or inches.  
This is done before the scale/zoom transform, after rotation

## **Examples**

APPLDATA 024 "LA=10, TA=50";

This command line adds 10 mm to the leading edge and 50mm to the trailing edge.

## **Behaviour**

LA, LR, TA and TR can all be defined simultaneously.

# — Framing (APPLDATA 025)

## — Syntax

**APPLDATA 025** “FM=FramingMode, AX=AreaX, AY= AreaY, AW=AreaWidth, AH=AreaHeight”;

parameter	meaning	type	default
<i>FramingMode</i>	either DISABLED(0) or DELETE THE INTERIOR(1) or the EXTERIOR(2) of the rectangle	<num>	0
<i>AreaX</i>	must be in the range 0 to 1219 mm/48 inch	<fxp>	0
<i>AreaY</i>	must be in the range 0 to 914 mm/36 inch	<fxp>	0
<i>AreaWidth</i>	must be in the range 0 to 914 mm/36 inch	<fxp>	0
<i>AreaHeight</i>	must be in the range 0 to 1219 mm/48 inch	<fxp>	0

[22] Framing (ID=025) syntax

## — Description

This command makes it possible to clip part of the plot, i.e. delete either a border around a rectangle in the middle or delete that middle rectangle while keeping the border.

**FM** This setting defines whether *framing* is DISABLED(0) or whether the INTERIOR(1) or the EXTERIOR(2) of the rectangle must be deleted.

**AX** Abscissa of the reference point of the area.

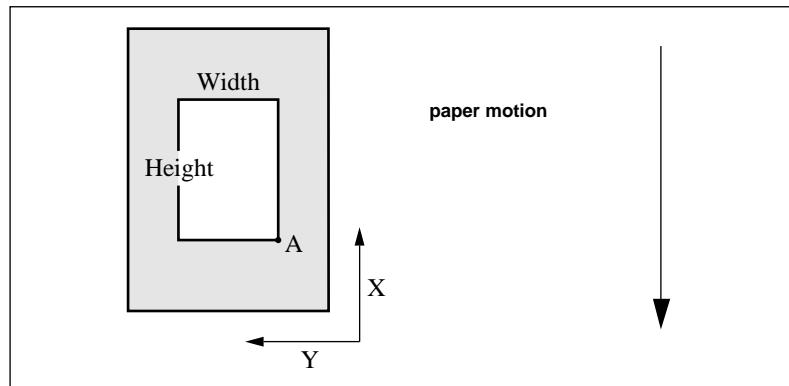
**AY** Ordinate of the reference point of the area.

**AW** Defines the *area width* from the reference point (AX,AY).

**AH** Defines the *area height* from the reference point (AX,AY).

The values are expressed in millimetres or inches depending on the units setting in the RCF header.

The X axis corresponds to the feed axis, the Y axis corresponds to the cross-feed axis. The axes used in the plot are not relevant.



[23] Definition of the corners of the rectangle

## Examples

APPLDATA 025 "FM=2, AX=130, AY= 90, AW=30, AH=40";

This command line only keeps a rectangle somewhere near the middle of an A4 drawing.

## Behaviour

If the values are defined without a *framing mode*, no operation is performed.

# Original to copy matrix (APPLDATA 026)

## Syntax

**APPLDATA 026** "S0=format, S1=format, S2=format, S3=format";

parameter	meaning	type	default
<i>format</i>	output format for the given input format: S0(0), S1(1), S2(2), S3(3), S4(4)	<num>	0

## Description

This command describes the *original to copy matrix*, which is used to select an output format that corresponds to the input format. It applies only when AutoScale and AutoFormat are both defined (see ‘Transformations (APPLDATA 003)’ on page 35 and ‘Media selection (APPLDATA 021)’ on page 41).

The following classes of formats are available:

	<i>DIN (European)</i>	<i>small ANSI (American)</i>	<i>large ANSI (American)</i>
<i>S0</i>	A0, 700 mm, 36"(E+)	30", 34"(E),	36" (E+)
<i>S1</i>	A1, 500 mm	22"(D)	24" (D+)
<i>S2</i>	A2	17"(C)	18" (C+)
<i>S3</i>	A3	11"(B)	12" (B+)
<i>S4</i>	A4	8.5"(A)	9" (A+)

By means of the *original to copy matrix* you can specify how files of a class of input formats will be printed. In other words, what output format class belongs to a particular input format.

**S0** This optional parameter specifies the output format (S0..S4), for the input format S0.

**S1** This optional parameter specifies the output format (S0..S4), for the input format S1.

**S2** This optional parameter specifies the output format (S0..S4), for the input format S2.

**S3** This optional parameter specifies the output format (S0..S4), for the input format S3.

**S4** This optional parameter specifies the output format (S0..S4), for the input format S4.

---

## Examples

This command line defines the matrix as indicated below:

APPLDATA 026 “S0=1, S1=1, S2=3, S3=3, S4=3”;

<i>Input format</i>	<i>output format</i>				
	S0	S1	S2	S3	S4
<i>S0</i>		x			
<i>S1</i>		x			
<i>S2</i>				x	
<i>S3</i>				x	
<i>S4</i>				x	

---

## Behaviour

The *original to copy matrix* only applies if the *auto scale* and *auto format* commands have both been defined.

If the system is configured in ANSI 8.5" *and* 9", the original to copy matrix always scales plots to the small series (8.5" series) format.

## Scan to file (APPLDATA 028)

### Syntax

**APPLDATA 028** “ CM=CoMmand, TS=TransferSize, FF=FileFormat, FO=FileOrganization, CP=ComPression, RS=ReSolution, JI=JobId, AI=AccountId, UI=UserId, TX=TeXtstring”;

parameter	meaning	type	default
<i>CoMmand</i>	ENABLE SCAN MODE (0), DISABLE SCAN MODE (1), ABORT SCAN JOB (2) START SCAN JOB (3), UPDATE SETTINGS (4), GET SCAN STATUS (5)	<num>	
<i>TransferSize</i>	Size of data blocks in bytes	<num>	65024
<i>FileFormat</i>	Format of the raster image: TIFF (0) or CALS (1)	<num>	0
<i>FileOrganization</i>	RAW (0), TILED (1), STRIPPED (2)	<num>	0
<i>ComPression</i>	Compression Type: NONE (0), GROUP4 (1), GROUP 3 1D (2), GROUP 3 2D (3)	<num>	0
<i>ReSolution</i>	200 or 400 dpi	<num>	400
<i>JobId</i>	Host Job Identification	<num>	
<i>AccountId</i>	Host Account Identification	<num>	
<i>UserId</i>	Host User Identification	<num>	
<i>TeXtstring</i>	User string sent to scanner	<str>	

---

## Description

This appldata is used to control the scan-to-file functionality. It is only allowed in RCF headers of type 'ScanToFile'.

**CM** This parameter changes the controller's state.

- Enable scan-mode (0)

indicates that the host is ready to receive data, using the transfer size indicated by the TS parameter. The transfer size setting will be used until the scan-mode is disabled, and cannot be changed in-between.

- Disable scan-mode (1)

will make the controller switch back to normal printing operation after all pending scans are uploaded.

- Abort scan-job (2)

will abort the scan-job whose ID is given, using the JI parameter.

- Start scan-job (3)

actually starts the scan. It must be accompanied by the JI parameter, and optionally by the other parameters like FileFormat etc.

- Update settings (4)

can be used to change the scan parameters of the scan job (specified by JI) once the scan-mode is enabled.

- Get scan status (5)

can be used to ask the controller to send back status information using APPLDATA 106.

**TS** This parameter specifies the transfer block size in bytes. The transfer block size is the total size of the data block as sent from the controller to the host, and must be a multiple of 512 bytes.

**FF** This parameter defines the format of the image data, returned by the controller. Supported are TIFF and CALS type1.

**FO** Indicates how the raster data is organized:

- One contiguous block: RAW (0)

- Divided into tiles: TILED (1)

- Several scanned lines in a strip: STRIPPED (2)

**CP** This parameter specifies the compression method to use:

- NONE (0)
- GROUP4 (1)
- GROUP 3 1D (2)
- GROUP 3 2D (3)

**RS** This parameter specifies the resolution of the image, as it will be uploaded to the host.

**JI** Job Id, specifies the host identification for a scan job. It can be used later on to refer to a scan job, for example when cancelling it or to modify its settings.

**AI** Account Id, specifies the host's accounting identification for a scan job.

**UI** User Id, specifies the host's user identification for a scan job.

**TX** A user definable text string (max. 26 characters) that is used to identify the scan. It is displayed on the scanner's control panel. Since the string is part of the RCF, it may not contain commas or quotes.

## Examples

Enable the scan mode, and ask for 65024 byte transfer size:

```
BEGMF "";
MFVERSION 1;
MFDESC "Oce RCF, Version=2.3, Type=ScanToFile, Unit=Inches";
APPLDATA 028 "CM=0, TS=65024";
ENDMF;
```

When in scan mode, start an A0 scan, tiled TIFF group4, 400dpi, JobId 345, AccountId 4213, UserId 893:

```
BEGMF "";
MFVERSION 1;
MFDESC "Oce RCF, Version=2.3, Type=ScanToFile, Unit=Inches";
APPLDATA 028 "CM=3, PF=0, FO=1, FF=0, CP=1, RS=400, JI=345,
AI=4213, UI=893, TX=city_map";
ENDMF;
```

# Alignment (APPLDATA 029)

## Syntax

**APPLDATA 029** “LR=LeftRightalignment, TB=TopBottomalignment”;

parameter	meaning	type	default
<i>LeftRightalignment</i>	LEFT (0), RIGHT (1), CENTERED (2)	<num>	0
<i>TopBottomalignment</i>	TOP (0), BOTTOM(1), CENTERED (2)	<num>	0

[24] Alignment (ID=029) syntax

## Description

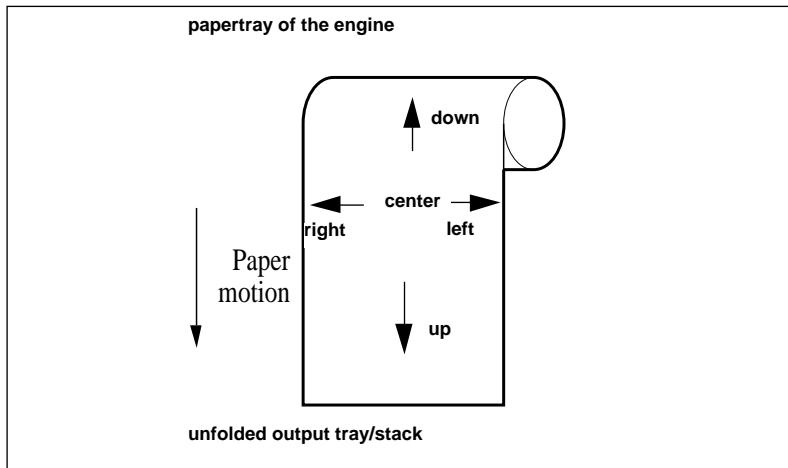
This function controls the position of the image on the copy material.

**LR** The bitmap may be *aligned* to the LEFT(0), to the RIGHT(1) edge, or it may be CENTERED(2).

**TB** The bitmap may be *aligned* to the TOP(0), to the BOTTOM(1) edge, or it may be CENTERED(2).

Five combinations are possible: top-left, top-right, bottom-left, bottom-right and centered. Where,

- top-left is LR(0), TB(0)
- top-right is LR(1), TB(0)
- bottom-left is LR(0), TB(1)
- bottom-right is LR(1), TB(1)
- centered is LR(2), or TB(2). Either LR is 2 , or TB is 2, or both LR and TB are 2.



[25] Alignment options

## Example

APPLDATA 029 “LR=0, TB=0”;

The bitmap will be placed on the left, on the top edge.

For bidsets you should set standard cut, zoom 100%, and fixed output size. If so, you can use the left and bottom alignment option:

```

BEGMF "RCH";
MFVERSION 1;
MFDESC "Oce RCF, Version=2.0,
Type=Header, Unit=Metric";
APPLDATA 002 "JB=1,CM=1,CO=2";
APPLDATA 003 "XS=1, YS=1";
APPLDATA 021 "ME=0, PF=6";
APPLDATA 023 "CI=1";
APPLDATA 029 "LR=0, TB=0";
ENDMF;

PLOT file
TIFF

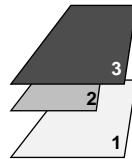
PLOT file
HP-GL

BEGMF "RCH";
MFVERSION 1;
MFDESC "Oce RCF, Version=2.0,
Type=Header, Unit=Metric";
APPLDATA 002 "JB=2";
ENDMF;

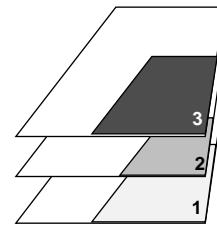
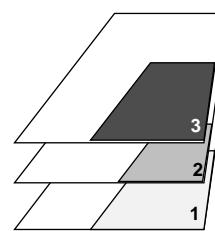
PLOT file
TIFF

```

input:



output:



[26] Bidset example

---

---

## HP-GL emulation (APPLDATA 050)

---

### Syntax

**APPLDATA 050** “**TY**=plotterTYpe, **OR**=ORigin, **SP**=SP0\_eof”;

parameter	meaning	type	default
<i>plotterTYpe</i>	type of emulated HP plotter (see below)	<num>	7
<i>ORigin</i>	plot origin, LL(0), LR(1), UL(2), UR(3) or Center(4)	<num>	4
<i>SP0_eof</i>	pen 0 indicates end-of-file, YES (0) or NO (1)	<num>	1

[27] HP-GL emulation (ID=050) syntax

---

### Description

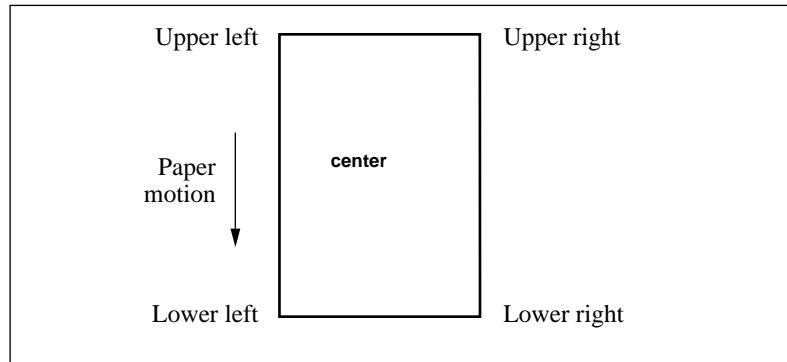
This command is used to specify the HP-GL parameter.

**TY** The *type* (model number) of plotter that should be emulated. This is used for HP-GLs OI (Output Identification) command.

type	model
0	HP-7440
1	HP-7475A
2	HP-7550A
3	HP-7570A
4	HP-7580A
5	HP-7580B
6	HP-7585B
7	HP-7586B
8	HP-7595A
9	HP-7596A
10	HP-7600
11	DesignJet

[28] HP-GL emulations

**OR** The *origin* of the plot: LOWER LEFT(0), LOWER RIGHT(1), UPPER LEFT(2), UPPER RIGHT(3) or CENTER(4).



[29] Plot origin options

**SP** If ENABLED, the selection of pen 0 is interpreted as EOForiginal (end of file). There is no pen #0 in HP-GL and some HP-GL files use the SP0 command to indicate that the plot is finished. If DISABLED, pen 0 behaves like a default HP-GL pen.

## Example

The next command will set the origin to lower left, and pen 0 as end of file:

```
APPLDATA 050 "OR=0, SP=0";
```

## HP-GL/2 emulation (APPLDATA 051)

### Syntax

**APPLDATA 051** “TY=plotterTYpe, OR= ORigin, SP=SP0\_eof,  
PP=PenPriority”;

<b>parameter</b>	<i>meaning</i>	<i>type</i>	<i>default</i>
<i>plotterTYpe</i>	type of emulated HP plotter (see below)	<num>	11
<i>ORigin</i>	plot origin, LL(0), LR(1), UL(2), UR(3), CENTER(4)	<num>	1
<i>SP0_eof</i>	pen 0 indicates end-of-file YES(0) or NO(1)	<num>	1
<i>PenPriority</i>	language(0) or remote(1)	<num>	0

[30] HP-GL/2 emulation (ID=051) syntax

### Description

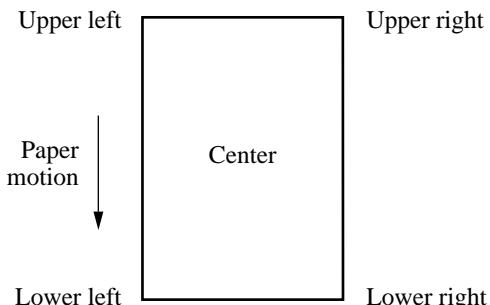
This command is used to specify HP-GL/2 related parameters. HP-RTL is included into the HP-GL/2 emulation.

**TY** The type (model number) of plotter that should be emulated. This is used for HP-GLs OI (Output Identification) command.

<i>type</i>	<i>model</i>
8	HP-7595B
9	HP-7596A
10	HP-7600
11	DesignJet

[31] HP-GL/2 emulation options

**OR** The origin of the plot, LOWER LEFT(0), LOWER RIGHT(1), UPPER LEFT(2), UPPER RIGHT(3), CENTER(4).



[32] Plot origin options

**SP** If ENABLED, the selection of pen 0 is interpreted as EOF (end of file). HP-GL/2 files may use the SP0 command to indicate that the plot is finished. If DISABLED, pen 0 behaves like a default HP-GL/2 pen.

**PP** In HP-GL/2, all pens are defined by default or in the plotfile. This is the default behaviour, i.e. PEN PRIORITY set to LANGUAGE(0). If the pens as defined with the help of APPLDATA 01 must be used instead, *pen priority* should be set to REMOTE(1), i.e. use the RCH instead of the language defaults/settings.

---

## Example

This command line will set the origin to lower right and set the emulated plotter to DesignJet:

APPLDATA 051 “TY=11, OR=1”;

# ASCII emulation (APPLDATA 056)

## Syntax

**APPLDATA 056** “EL=EndofLine, LO=LineOverflow, FS=FontSize, TM=TopMargin, BM=BottomMargin, LM=LeftMargin, RM=RightMargin, OR=ORIENTATION”;

parameter	meaning	type	default
<i>EndofLine</i>	type of line break in use, CR-ONLY(0) LF-ONLY(1) CR-LF(2)	<num>	0
<i>LineOverflow</i>	long line handling, WRAP(0) or TRUNCATE(1)	<num>	0
<i>FontSize</i>	font size to use: 8, 10 or 12 points	<num>	10 points
<i>TopMargin</i>	allowed range: 0 to 25mm/1inch	<fxp>	10 mm 0.4 inch
<i>BottomMargin</i>	allowed range: 0 to 25mm/1inch	<fxp>	10 mm 0.4 inch
<i>LeftMargin</i>	allowed range: 0 to 25mm/1inch	<fxp>	10 mm 0.4 inch
<i>RightMargin</i>	allowed range: 0 to 25mm/1inch	<fxp>	10 mm 0.4 inch
<i>ORIENTATION</i>	direction, PORTRAIT(0) LANDSCAPE(1)	<num>	0

[33] ASCII (ID=056) syntax

## Description

This command describes the ASCII parameters.

**EL** This parameter defines the type of line break the plot will be processed with: CARRIAGE RETURN(0), LINE FEED(1) or BOTH(2).

**LO** This setting specifies whether the line is TRUNCATED(1) to the page width or if a line feed is added so that the remaining characters are WRAPPED(2) to the next line.

**FS** This is the size of the font that is used, expressed in points.

**TM** The margins around the text must be in the following range: 0..25mm/1inch. TOP is related to the input of the paper, not to the orientation.

**BM** The margins around the text must be in the following range: 0..25mm/1inch. BOTTOM is related to the input of the paper, not to the orientation.

**LM** The margins around the text must be in the following range: 0..25mm/1inch. LEFT is related to the input of the paper, not to the orientation.

**RM** The margins around the text must be in the following range: 0..25mm/1inch. RIGHT is related to the input of the paper, not to the orientation.

**OR** The orientation of the file, Portrait or Landscape.

## Example

This command line selects a 10 point font, wraps long lines and uses a carriage return as line break.

APPLDATA 056 “EL=0, LO=0, FS=10”;

**Note:** You must always identify ASCII files by specifying ACCII as language parameter of APPLDATA 059. Automatic language sensing can not detect ASCII files (see ‘Automatic language sensing (APPLDATA 059)’ on page 69).

## CalComp emulation (APPLDATA 058)

### Syntax

**APPLDATA 058** “OR=ORigin, ST=STep, CK=ChecKsum,  
EM=EndofMessage, SY=SYnc\_code, DB=DouBle\_sync, PP=PenPriority”;

parameter	meaning	type	default
<i>ORigin</i>	plot origin, LL(0), LR(1), UL(2), UR(3) or CENTER(4)	<num>	1
<i>STep</i>	steps per inch: 100, 200, 400, 500, 1016, 2032 or 4064	<num>	400
<i>ChecKsum</i>	check sum YES(0) or NO(1)	<num>	0
<i>EndofMessage</i>	character flagging the end of the data sequence: 0 <= eom <=30	<num>	3
<i>SYnc_code</i>	character flagging the end of a block of plot data: 0 <= sync code <=63	<num>	2
<i>DouBle_sync</i>	double sync YES(0) or NO(1)	<num>	0
<i>PenPriority</i>	LANGUAGE(0) or REMOTE(1)	<num>	0

[34] Calcomp emulation (ID=058) syntax

### Description

This command is used to select the CalComp emulation and to define the parameters for the CalComp 906/907 languages.

**OR** The *origin* of the plot: LOWER LEFT(0), LOWER RIGHT(1), UPPER LEFT(2), UPPER RIGHT(3) or CENTER(4).

**ST** Number of *steps* per inch. This parameter should correspond to the resolution of the plot.

**CK** Enables or disables the *checksum* mechanism.

**EM** Character code for the *end-of-message* byte, indicating the end of a data sequence.

**SY** Character code for the *sync* byte, indicating the beginning of the plot data.

**DB** Enables or disables the *double sync*, i.e. whether or not the byte which begins the plot data is sent twice.

**PP** Parameter to select the pens defined in the plot file: LANGUAGE(0) or to select the pens as defined in the RCH: REMOTE(1).

---

## Example

The following command will set the step to 2032 steps/inch, origin LOWER RIGHT, sync byte to 0 and eom byte to 30 (decimal):

APPLDATA 058 “OR=1, ST=2032, SY=0, EM=30”;

---

## Automatic language sensing (APPLDATA 059)

---

### Syntax

**APPLDATA 059** “LG=LanGuage, WS=alsWorkingSet”;

parameter	meaning	type	default
<i>LanGuage</i>	the language to process or AUTO(0)	<num>	0
<i>alsWorkingSet</i>	If in auto-mode, the set of languages the ALS may choose from.	<lst>	

[35] Automatic language sensing (ID=059) syntax

---

### Description

This command configures the *automatic language sensing* feature. The language are listed as follows.

number	language
0	auto
1	HP-GL
2	HP-GL/2 & HP-RTL
3	CALS
6	TIFF
7	ASCII
12	CALCOMP

[36] Available languages

**LG** Parameter to set the language. If set to AUTO, *automatic language sensing* is used with the set of languages specified by the WS parameter.

**WS** The list of languages from which the ALS may choose.

**Note:** *Automatic language sensing can not detect ASCII files.*

## **Example**

The following APPLDATA configures the *automatic language sensing (ALS)* for HP-GL/2 and TIFF, and activates it:.

APPLDATA 059 “WS=<2,6>, LG=0”;

## Scanner info/data (APPLDATA 106)

### Syntax

**APPLDATA 106** “US=UploadStatus, TY=TYpe, DL=DataLength,  
WD=WiDth, LG=LenGth, JI=JobId, EC=ErrorCode”;

parameter	meaning	type	default
<i>UploadStatus</i>	CONTROLLER IS BUSY (0), READY TO SCAN (1), SCANNER NOT AVAILABLE (2) ERROR DETECTED (3) SCANNER PROCESSING (4) JOB ABORTED (5) JOB UPDATED (6)	<num>	0
<i>TYpe</i>	NO DATA (0), DATA BLOCK (1) DATA END (2)	<num>	0
<i>DataLength</i>	Number of bytes of image data in block	<num>	
<i>WiDth</i>	Width of scanned image in pixels	<num>	
<i>LenGth</i>	Length of the scanned image in pixels	<num>	
<i>JobId</i>	Job identification as defined by the host in the scan request	<num>	
<i>ErrorCode</i>	JOB NOT FOUND (0) RCF_INVALID_PARAM (2) RCF_MISSING_PARAM (3) RCF_SYNTAX_ERROR (4) MEMORY FULL (5) COMPRESSION OVERFLOW (6) CONTROLLER ERROR (7)	<num>	

[37] Scanner info/data (ID=106) syntax

---

## Description

This APPLDATA returns the scanned data or status information to the host. It is always sent in response to a Scan to file command (see ‘Scan to file (APPLDATA 028)’ on page 55) and optionally followed by scanned data.

**US** This parameter is used to both inform the host about the status of the scan to file process, and to acknowledge a received command:

*Scan Status (in response to a Get Scan Status):*

- Controller is busy(0)  
is returned when the controller is busy, and not yet ready to enter scan mode.
- Ready to scan (1)  
means that the controller is ready to receive a ‘Start Scan Job’ command.
- Scanner not available (2)  
is returned when the controller is processing ‘Start Scan Job’ command(s), and is not able to accept new requests.

*Acknowledgement (in response to any other command. JI indicates the JobId to which it pertains):*

- Error Detected (3)  
The parameter EC contains the error code, explaining the kind of error.
- Scanner Processing (4)  
indicates that a scan is in progress, the settings for this scan can no longer be modified.
- Job aborted (5)  
indicates the successful completion of an ‘Abort Scan Job’ command.
- Job updated (6)  
confirms the ‘Update Settings’ command.

**TY** specifies the type of data that follows this RCF header (the JI parameter specifies to which job this block corresponds).

- No data (0)  
is used when the packet does not transport any scan data.
- Data block (1)  
means that there is a block of raster data.
- Data end (2)  
means that it is the last data block for the given JobId.

**DL** The amount of image data (in bytes), included in the block.

**WD** The width of the image in pixels (this parameter is sent with the last data block of the image).

**LG** The length of the image in pixels (this parameter is sent with the last data block of the image).

**JI** specifies the ID of the scan job to which this packet pertains.

**EC** Error code (accompanying by JI):

■ Job not found (0)

A command has been received, but the specified scan job does not exist.

■ Update ignored (1)

The host sent an ‘Update Settings’ command for a job that has already been flagged as being processed (scanner\_processing msg.)

■ RCF\_invalid\_param (2)

An RCF parameter is invalid (and will be ignored).

■ RCF\_missing\_param (3)

A command has been sent, but one of the mandatory parameters is missing. The command will be ignored.

■ RCF\_syntax\_error (4)

The controller detected a syntax error in the RCF. The command will be ignored.

■ Memory full (5)

There is not enough memory in the controller to store the scanned data. The job, identified by JobId, will be aborted by the controller. No data will be sent to the host.

■ Compression overflow (6)

A memory overflow occurred during internal treatment of the raster data (e.g. during the compression). The job, identified by JobId, will be aborted by the controller. No data will be sent to the host.

■ Controller error (7)

Unexpected controller error. Means that the current scan job is aborted.

---

## Example

When the host receives:

```
BEGMF "...";
MFVERSION 1;
MFDESC "Oce RCF, Version=2.3, Type=ScanToFile, Unit=inches";
APPLDATA 106 "WD=12576, LG=18987, TY=2, DL=3207, JI=23";
```

ENDMF;  
<32000 bytes of image data>

it receives 32000 bytes, of which 3207 bytes (of scan job 23) will be read,  
which is an image of 12576x18987 pixels.

**Note:** *This APPLDATA is sent to the host for every 'APPLDATA 28' received.*

---

## Typical examples of RCF jobs

---

### Single file

```
BEGMF "Header file for printing a single file";
MFVERSION 1;
MFDESC "Oce RCF, Version=2.0, Type=Header, Unit=Metric";

% Specific pen settings %
APPLDATA 001 "PN=1, PW=0.25, PP=16";
APPLDATA 001 "PN=2, PW=0.35, PP=16";
APPLDATA 001 "PN=3, PW=0.50, PP=16";
APPLDATA 001 "PN=4, PW=0.70, PP=16";
APPLDATA 001 "PN=5, PW=1.00, PP=16";
APPLDATA 001 "PN=6, PW=0.15, PP=16";
APPLDATA 001 "PN=7, PW=1.50, PP=16";
APPLDATA 001 "PN=8, PW=1.25, PP=16";

% 3 copies sortwise of a single file which is no long plot %
APPLDATA 002 "JB=0, CM=0, CO=3, LP=0";

% No scaling, no rotation, no shift. %
APPLDATA 003 "XS=1.0, YS=1.0, AS=0, RO=0, SL=0, SR=0, SU=0, SD=0";

% No stamp %
APPLDATA 020 "SS=0";

% A2, plain paper, no manual feed, search fit %
APPLDATA 021 "PF=2, ME=0, AF=0, BY=0, RM=0";

% Fold, standard, length 297mm, width 210mm, no binding edge %
APPLDATA 022 "FO=1, ME=0, LE=297, WI=210, BE=0";

% Deliver on belt 1, and generate a standard size %
APPLDATA 023 "DE=1, CI=1";

% No leading or trailing edge correction %
APPLDATA 024 "LA=0, LR=0, TA=0, TR=0";

% No framing %
APPLDATA 025 "FM=0";

% HPGL specific settings: centred origin, SP0 is the end of the data file. %
APPLDATA 050 "OR=4, SP=0";

% Data file is hpgl (i.e. overrule ALS) %
APPLDATA 059 "LG=1";

ENDMF;

< A2 HPGL data file >
```

---

## Sets

An example of printing a set:

```
BEGMF "First header file for printing a set";
MFVERSION 1;
MFDESC "Oce RCF, Version=2.0, Type=Header, Unit=Metric";

% Specific pen settings %
APPLDATA 001 "PN=1, PW=0.25, PP=16";
APPLDATA 001 "PN=2, PW=0.35, PP=16";
APPLDATA 001 "PN=3, PW=0.50, PP=16";
APPLDATA 001 "PN=4, PW=0.70, PP=16";
APPLDATA 001 "PN=5, PW=1.00, PP=16";
APPLDATA 001 "PN=6, PW=0.15, PP=16";
APPLDATA 001 "PN=7, PW=1.50, PP=16";
APPLDATA 001 "PN=8, PW=1.25, PP=16";

% 5 copies setwise; start of set; no long plot %
APPLDATA 002 "JB=1, CM=1, CO=5, LP=0";

% No scaling, no rotation, no shift. %
APPLDATA 003 "XS=1.0, YS=1.0, AS=0, RO=0, SL=0, SR=0, SU=0, SD=0";

% No stamp %
APPLDATA 020 "SS=0";

% A0, plain paper, no manual feed, search fit %
APPLDATA 021 "PF=0, ME=0, AF=0, BY=0, RM=0";

% Fold, standard, length 297mm, width 210mm, no binding edge %
APPLDATA 022 "FO=1, ME=0, LE=297, WI=210, BE=0";

% Deliver on belt 1, and generate a standard size %
APPLDATA 023 "DE=1, CI=1";

% No leading or trailing edge correction %
APPLDATA 024 "LA=0, LR=0, TA=0, TR=0";

% No framing %
APPLDATA 025 "FM=0";

% HPGL specific settings: centred origin, SP0 is the end of the data file. %
APPLDATA 050 "OR=4, SP=0";

% Data file is hpgl (i.e. overrule ALS) %
APPLDATA 059 "LG=1";

ENDMF;

< A0 HPGL data file >
```

```

BEGMF "Header for second file of a set";
MFVERSION 1;
MFDESC "Oce RCF, Version=2.0, Type=Header, Unit=Metric";

% All default set context settings can be used, except the paper size. %
% Use A2 paper %
APPLDATA 021 "PF=2";

ENDMF;

< A2 HPGL data file >

BEGMF "header file for the third file of a set";
MFVERSION 1;
MFDESC "Oce RCF, Version=2.0, Type=Header, Unit=Metric";

% All default set context settings can be used, except paper size %
% and dataformat. %

% Use A1 paper size. %
APPLDATA 021 "PF=1";

% Data file is tiff (i.e. overrule ALS) %
APPLDATA 059 "LG=6";

ENDMF;

< A1 TIFF data file >

BEGMF "Last header file for printing a set";
MFVERSION 1;
MFDESC "Oce RCF, Version=2.0, Type=Header, Unit=Metric";

% The last file is printed with the default set context settings.%
% This header file only indicates the end of the set. %
APPLDATA 002 "JB=2";

ENDMF;

< A0 HPGL data file >

```

---

## Matrices

Example of a matrix job:

	Matrix Program 1	Matrix Program 2	Matrix Program 3
Original 1	x		x
Original 2		x	x
Original 3	x	x	x
Original 4		x	x
Original 5	x	x	x
Original 6	x		x

```
BEGMF "Matrix Program 1";
MFVERSION 1;
MFDESC "Oce RCF, Version=2.0, Type=MatrixPrg, Unit=Metric";

% 2 copies setwise of all the plots 1, 3, 5, and 6; all no long plots %
APPLDATA 002 "CM=1, CO=2, PL=<1,3,5,6>";

% No scaling, no rotation, no shift. %
APPLDATA 003 "XS=1.0, YS=1.0, AS=0, RO=0, AR=0, SL=0, SR=0, SU=0, SD=0";

% No stamp %
APPLDATA 020 "SS=0";

% Plain paper, autoformat enabled, no manual feed, search fit %
APPLDATA 021 "ME=0, AF=1, BY=0, RM=0";

% Fold and punch, standard way, length 297mm, width 210mm, no binding edge %
APPLDATA 022 "FO=2, ME=0, LE=297, WI=210, BE=0";

% Deliver on belt 1, and generate a standard size %
APPLDATA 023 "DE=1, CI=1";

% No leading or trailing edge correction %
APPLDATA 024 "LA=0, LR=0, TA=0, TR=0";

% No framing %
APPLDATA 025 "FM=0";

ENDMF;

BEGMF "Matrix Program 2";
MFVERSION 1;
MFDESC "Oce RCF, Version=2.0, Type=MatrixPrg, Unit=Metric";

% 1 copy sortwise of all the plots 2, 3, 4, and 5; all no long plots %
APPLDATA 002 "CM=0, CO=1, PL=<2,3,4,5>";

% Autoscale enabled, no rotation, no shift. %
```

```

APPLDATA 003 "XS=1.0, YS=1.0, AS=1, RO=0, AR=0, SL=0, SR=0, SU=0, SD=0";
% No stamp %
APPLDATA 020 "SS=0";

% A3 Plain paper, no manual feed, search fit %
APPLDATA 021 "PF=3, ME=0, AF=0, BY=0, RM=0";

% Unfolded %
APPLDATA 022 "FO=0";

% Deliver in the stacker, and generate a standard size %
APPLDATA 023 "DE=0, CI=1";

% No leading or trailing edge correction %
APPLDATA 024 "LA=0, LR=0, TA=0, TR=0";

% No framing %
APPLDATA 025 "FM=0";

ENDMF;

BEGMF "Matrix Program 3";
MFVERSION 1;
MFDESC "Oce RCF, Version=2.0, Type=MatrixPrg, Unit=Metric";

% 5 copies setwise of all the plots %
APPLDATA 002 "CM=1, CO=5, PL=<1,2,3,4,5,6>";

% No scaling, no rotation, no shift. %
APPLDATA 003 "XS=1.0, YS=1.0, AS=0, RO=0, AR=0, SL=0, SR=0, SU=0, SD=0";

% Use stamp nr 1, at the bottom, in a small, black font %
APPLDATA 020 "SS=1, SN=1, PO=0, FS=0, GS=0";

% Plain paper, autoformat enabled, no manual feed, search fit %
APPLDATA 021 "ME=0, AF=1, BY=0, RM=0";

% Fold, standard way, length 297mm, width 210mm, no binding edge %
APPLDATA 022 "FO=1, ME=0, LE=297, WI=210, BE=0";

% Deliver on belt 2, and generate a standard size %
APPLDATA 023 "DE=2, CI=1";

% No leading or trailing edge correction %
APPLDATA 024 "LA=0, LR=0, TA=0, TR=0";

% No framing %
APPLDATA 025 "FM=0";

ENDMF;

BEGMF "Header file for file 1 of the matrix";
MFVERSION 1;

```

```

MFDESC "Oce RCF, Version=2.0, Type=Header, Unit=Metric";

% Specific pen settings %
APPLDATA 001 "PN=1, PW=0.25, PP=16";
APPLDATA 001 "PN=2, PW=0.35, PP=16";
APPLDATA 001 "PN=3, PW=0.50, PP=16";
APPLDATA 001 "PN=4, PW=0.70, PP=16";
APPLDATA 001 "PN=5, PW=1.00, PP=16";
APPLDATA 001 "PN=6, PW=0.15, PP=16";
APPLDATA 001 "PN=7, PW=1.50, PP=16";
APPLDATA 001 "PN=8, PW=1.25, PP=16";

% Start of set; plot number 1 %
APPLDATA 002 "JB=1, PN=1";

% HPGL specific settings: centred origin, SP0 is the end of the data file. %
APPLDATA 050 "OR=4, SP=0";

% Data file is hpgl (i.e. overrule ALS) %

APPLDATA 059 "LG=1";

ENDMF;

< A0 HPGL data file >

BEGMF "Header file for file 2 of the matrix";
MFVERSION 1;
MFDESC "Oce RCF, Version=2.0, Type=Header, Unit=Metric";
% Specific pen settings %
APPLDATA 001 "PN=1, PW=0.25, PP=16";
APPLDATA 001 "PN=2, PW=0.35, PP=16";
APPLDATA 001 "PN=3, PW=0.50, PP=16";
APPLDATA 001 "PN=4, PW=0.70, PP=16";
APPLDATA 001 "PN=5, PW=1.00, PP=16";
APPLDATA 001 "PN=6, PW=0.15, PP=16";
APPLDATA 001 "PN=7, PW=1.50, PP=16";
APPLDATA 001 "PN=8, PW=1.25, PP=16";

% Plot number 2 %
APPLDATA 002 "PN=2";

% CalComp specific settings: centred origin, stepsize 2032 steps/inch %
% checksum yes, end of message character 13, synch character 22, %
% double checksum, pen priority remote. %
APPLDATA 058 "OR=4, ST=2032, CK=0, EM=13, SY=22, DB=0, PP=1";

% Data file is calcomp (i.e. overrule ALS) %
APPLDATA 059 "LG=12";

ENDMF;

< A1 CalComp data file >

BEGMF "Header file for file 3 of the matrix";
MFVERSION 1;

```

```

MFDESC "Oce RCF, Version=2.0, Type=Header, Unit=Metric";

% Plot number 3 %
APPLDATA 002 "PN=3";

% Data file is tiff (i.e. overrule ALS) %
APPLDATA 059 "LG=6";

ENDMF;

< A0 TIFF data file >

BEGMF "Header file for file 4 of the matrix";
MFVERSION 1;
MFDESC "Oce RCF, Version=2.0, Type=Header, Unit=Metric";

% Specific pen settings %
APPLDATA 001 "PN=1, PW=0.35, PP=16";
APPLDATA 001 "PN=2, PW=0.50, PP=16";

% Plot number 4 %
APPLDATA 002 "PN=4";

% CalComp specific settings: centred origin, stepsize 2032 steps/inch %
% checksum yes, end of message character 13, synch character 22, %
% double checksum, pen priority remote. %
APPLDATA 058 "OR=4, ST=2032, CK=0, EM=13, SY=22, DB=0, PP=1";

% Data file is calcomp (i.e. overrule ALS) %
APPLDATA 059 "LG=12";

ENDMF;

< A2 CalComp file >

BEGMF "Header file for file 5 of the matrix";
MFVERSION 1;
MFDESC "Oce RCF, Version=2.0, Type=Header, Unit=Metric";

% Specific pen settings %
APPLDATA 001 "PN=1, PW=0.25, PP=16";
APPLDATA 001 "PN=2, PW=0.35, PP=16";
APPLDATA 001 "PN=3, PW=0.50, PP=16";
APPLDATA 001 "PN=4, PW=0.70, PP=16";
APPLDATA 001 "PN=5, PW=1.00, PP=16";
APPLDATA 001 "PN=6, PW=0.15, PP=16";
APPLDATA 001 "PN=7, PW=1.50, PP=16";
APPLDATA 001 "PN=8, PW=1.25, PP=16";

% plot number 5 %
APPLDATA 002 "PN=5";

% HPGL specific settings: centred origin, SP0 is the end of the data file. %
APPLDATA 050 "OR=4, SP=0";

% Data file is hpgl (i.e. overrule ALS) %

```

```

APPDATA 059 "LG=1";

ENDMF;

< A3 HPGL data file >

BEGMF "Header file for file 6 of the matrix";
MFVERSION 1;
MFDESC "Oce RCF, Version=2.0, Type=Header, Unit=Metric";

% Specific pen settings %
APPDATA 001 "PN=1, PW=0.25, PP=16";
APPDATA 001 "PN=2, PW=0.40, PP=16";
APPDATA 001 "PN=3, PW=1.00, PP=16";

% end of set; plot number 6 %
APPDATA 002 "JB=2, PN=6";

% HPGL specific settings: centred origin, SP0 is the end of the data file. %
APPDATA 050 "OR=4, SP=0";

% Data file is hpgl (i.e. overrule ALS) %
APPDATA 059 "LG=1";

ENDMF;

< A4 HPGL data file >

```

# APPLDATA Quick reference table

	Meaning	Type	Default
<b>APPLDATA 001 Pen attributes</b>			
<i>PenNumber</i>	a single pen specified by a pen number, 1..999 range of pens: n1-n2, 1 <= n1 <= n2 <= 999	<num> <n1>-<n2>	
<i>PenWidth</i>	pen width in millimetres or inches: 0.12mm or 0.025" <fxp> -16.25 mm/0.639 inch at 400 dpi		
<i>PenPattern</i>	pen pattern: 1..16 are grey scales, 17..32 are Océ patterns	<num>	16 (black)
<b>APPLDATA 002 Job parameters</b>			
<i>Copies</i>	number of copies to print, 1..999	<num>	1
<i>AccountId</i>	account number (max 9 digits)	<num>	0
<i>UserId</i>	user number (max 9 digits)	<num>	0
<i>JobBoundary</i>	SINGLE FILE(0), START OF SET(1) or END OF SET(2)	<num>	0
<i>CopyMethod</i>	sort copies BY PAGE(0) ( <i>Only if JB=0!</i> ) or BY SET(1)	<num>	0
<i>PlotList</i>	list of plots for the specified matrix program	<lst>	0
<i>PlotNumber</i>	number of the plot inside the matrix job	<num>	0
<b>APPLDATA 003 Transformations</b>			
<i>XScale</i>	X scale (or zoom) factor 0.2500..4.0000	<fxp>	1
<i>YScale</i>	Y scale (or zoom) factor 0.2500..4.0000	<fxp>	1
<i>ROtation</i>	rotation angle: 0, 90,180 or 270 degrees.	<num>	0
<i>Mirror</i>	Drawing can be mirrored in the X direction: NO MIRROR (0) or MIRROR IN X (1)	<num>	0
<i>AutoScale</i>	automatic scale or zoom, DISABLED(0) or ENABLED(1)	<num>	0
<i>AutoRotate</i>	automatic rotation, DISABLED(0), FOLDING (1) or PRODUCTIVE(2)	<num>	0
<i>LegendControl</i>	legenda control, DISABLED(0), or ENABLED(1)	<num>	0
<i>ShiftUp</i>	shift up factor: 0 to 1219mm/48inch	<fxp>	0
<i>ShiftDown</i>	shift down factor: 0 to 1219mm/ 48inch	<fxp>	0
<i>ShiftLeft</i>	shift left factor: 0 to 914mm/36inch	<fxp>	0
<i>ShiftRight</i>	shift right factor: 0 to 914mm/ 36inch	<fxp>	0

	Meaning	Type	Default
<b>APPLDATA 020 Stamp</b>			
<i>StampSelection</i>	stamping ENABLED(1) or DISABLED(0)	<num>	0
<i>StringNumber</i>	predefined string number, from 1 to 50	<num>	1
<i>Position</i>	predefined stamp position: BOTTOM(0), MIDDLE(1) or TOP(2)	<num>	1
<i>FontSize</i>	predefined font: SMALL (0) or LARGE (1)	<num>	0
<i>GreyScale</i>	predefined grey scale choice: 0, 1, 2, 3 black = 0, darkgrey = 1, grey = 2, lightgrey = 3	<num>	0
<b>APPLDATA 021 Media selection</b>			
<i>PaperFormat</i>	format of the paper: A0, A1 ... (from 0/A0 to 21/700_MM)	<num>	AutoFormatresult
<i>Media</i>	medium choice: PAPER(0), TRANSPARENT(1) or POLYESTER(2)	<num>	0
<i>AutoFormat</i>	automatic format ENABLED(1) or DISABLED(0)	<num>	0
<i>ByPass</i>	manual feed ENABLED(1) or DISABLED(0)	<num>	0
<i>RolloverrulingMethod</i>	exactly the FIT (0), NEXT SIZE(1) or REDUCED(2)	<num>	0
<b>APPLDATA 022 Finishing</b>			
<i>FoldOptions</i>	FOLD OFF(0), FOLD ON(1) or FOLD AND PUNCH(2)	<num>	0
<i>foldMEthod</i>	STANDARD(0), ERICSSON(1) or AFNOR(2)	<num>	0
<i>Length</i>	fold packet length from 276mm/10.9 inch to 310mm/12.2inch	<fxp>	297mm or 11"
<i>Width</i>	fold packet width from 186mm/7.3 inch to 230mm/9inch	<fxp>	210mm or 8.5"
<i>BindingEdge</i>	NONE(0) or a value in the range 15mm/0.6inch to 30mm/1.2inch	<fxp>	0
<i>FoldOrientation</i>	AUTO(0), or PORTRAIT(1)	<num>	0
<b>APPLDATA 023 Delivery</b>			
<i>CutInfo</i>	cut plot on the size of the PLOT(0) on the STAND- ARD SIZE(1), or on CUSTOM CUT (2)	<num>	0
<i>CutLength</i>	cut length 210 to 15000 mm 85 to 6000 0.1 inch	<num>	210 mm 8.5 inch
<i>Output</i>	If Lower Container Unit installed: choosing upper or lower output to deliver plot: UPPER(0), LOWER(1) or AUTO(2)	<num>	0

	Meaning	Type	Default
<i>DE</i> posit	If Folder with High Capacity Output Unit installed: de-<num> posit plot in the STACKER(0), on BELT1(1), on BELT2(2) or on EITHER BELT(3)		0
	If High Capacity Stacker installed: CONTINUOUS (4) PER SET (5) PER JOB (6) BIN NUMBER (7)		
<i>BinNumber</i>	If High Capacity Stacker installed, and BIN NUMBER is selected (DE=7): All copies are delivered onto the specified bin: 1..6	<num>	1
<i>JOG</i> ging	If High Capacity Stacker installed: Jogging (offset stack) DISABLED(0) ENABLED(1)	<num>	0

---

#### APPLDATA 024 Edge correction

<i>LeadingedgeAdd</i>	0 to 297.4mm/11inch are added to the leading edge	<fxp>	0
<i>LeadingedgeRemove</i>	0 to 100mm/4inch are removed from the leading edge	<fxp>	0
<i>TrailingedgeAdd</i>	0 to 297.4mm/11inch are added to the trailing edge	<fxp>	0
<i>TrailingedgeRemove</i>	0 to 100mm/4inch are removed from the trailing edge	<fxp>	0

---

#### APPLDATA 025 Framing

<i>FramingMode</i>	either DISABLED(0) or DELETE THE INTERIOR(1) or <num> the EXTERIOR(2) of the rectangle		0
<i>AreaX</i>	must be in the range 0 to 1219 mm/48 inch	<fxp>	0
<i>AreaY</i>	must be in the range 0 to 914 mm/36 inch	<fxp>	0
<i>AreaWidth</i>	must be in the range 0 to 914 mm/36 inch	<fxp>	0
<i>AreaHeight</i>	must be in the range 0 to 1219 mm/48 inch	<fxp>	0

---

#### APPLDATA 026 Original to copy matrix

<i>format</i>	output format for the given input format: S0(0), S1(1), S2(2), S3(3), S4(4)	<num>	0
---------------	--	-------	---

	Meaning	Type	Default
<b>APPLDATA 028 Scan to file</b>			
<i>CoMmand</i>	ENABLE SCAN MODE (0) DISABLE SCAN MODE (1) ABORT SCAN JOB (2) START SCAN JOB (3) UPDATE SETTINGS (4) GET SCAN STATUS (5)	<num>	0
<i>TransferSize</i>	Size of data blocks in bytes	<num>	65024
<i>FileFormat</i>	Format of the raster image: TIFF (0) or CALS (1)	<num>	0
<i>FileOrganization</i>	RAW (0), TILED (1), STRIPPED (2)	<num>	0
<i>ComPression</i>	Compression Type: NONE (0), GROUP4 (1), GROUP 3 1D (2), GROUP 3 2D (3)	<num>	0
<i>ReSolution</i>	200 or 400 dpi	<num>	400
<i>JobId</i>	Host Job Identification	<num>	
<i>AccountId</i>	Host Account Identification	<num>	
<i>UserId</i>	Host User Identification	<num>	
<i>TeXtstring</i>	User string sent to scanner	<str>	
<b>APPLDATA 029 Alignment</b>			
<i>LeftRightalignment</i>	LEFT (0), RIGHT (1), CENTRED (2)	<num>	0
<i>TopBottomalignment</i>	TOP (0), BOTTOM(1), CENTRED (2)	<num>	0
<b>APPLDATA 050 HP-GL emulation</b>			
<i>plotterTYpe</i>	type of emulated HP plotter (see below)	<num>	7
<i>ORigin</i>	plot origin, LL(0), LR(1), UL(2), UR(3) or Centre(4)	<num>	4
<i>SP0_eof</i>	pen 0 indicates end-of-file, YES (0) or NO (1)	<num>	1
<b>APPLDATA 051 HP-GL/2 emulation</b>			
<i>plotterTYpe</i>	type of emulated HP plotter (see below)	<num>	11
<i>ORigin</i>	plot origin, LL(0), LR(1), UL(2), UR(3), CENTRE(4)	<num>	1
<i>SP0_eof</i>	pen 0 indicates end-of-file YES(0) or NO(1)	<num>	1
<i>PenPriority</i>	language(0) or remote(1)	<num>	0

	Meaning	Type	Default
<b>APPLDATA 056 ASCII emulation</b>			
<i>EndofLine</i>	type of line break in use, CR-ONLY(0) LF-ONLY(1) CR-LF(2)	<num>	0
<i>LineOverflow</i>	long line handling, WRAP(0) or TRUNCATE(1)	<num>	0
<i>FontSize</i>	font size to use: 8, 10 or 12 points	<num>	10
<i>TopMargin</i>	allowed range: 0 to 25mm/1inch	<fxp>	10 mm 0.4 inch
<i>BottomMargin</i>	allowed range: 0 to 25mm/1inch	<fxp>	10 mm 0.4 inch
<i>LeftMargin</i>	allowed range: 0 to 25mm/1inch	<fxp>	10 mm 0.4 inch
<i>RightMargin</i>	allowed range: 0 to 25mm/1inch	<fxp>	10 mm 0.4 inch
<i>ORientation</i>	direction,PORTRAIT(0) LANDSCAPE(1)	<num>	0
<b>APPLDATAS 058 CalComp emulation</b>			
<i>ORigin</i>	plot origin, LL(0), LR(1), UL(2), UR(3) or CEN-TRE(4)	<num>	1
<i>STep</i>	steps per inch: 100, 200, 400, 500, 1016, 2032 or 4064	<num>	400
<i>ChecKsum</i>	check sum YES(0) or NO(1)	<num>	0
<i>EndofMessage</i>	character flagging the end of the data sequence: 0 <= <num> eom <=30	<num>	3
<i>SYnc_code</i>	character flagging the end of a block of plot data: 0 <=sync code <=63	<num>	2
<i>DouBle_sync</i>	double sync YES(0) or NO(1)	<num>	0
<i>PenPriority</i>	LANGUAGE(0) or REMOTE(1)	<num>	0
<b>APPLDATA 059 Automatic language sensing</b>			
<i>LanGuage</i>	the language to process or AUTO(0)	<num>	0
<b>APPLDATA 106 Scanner info/data</b>			
<i>UploadStatus</i>	CONTROLLER IS BUSY (0), READY TO SCAN (1), SCANNER NOT AVAILABLE (2) ERROR DETECTED (3) SCANNER PROCESSING (4) JOB ABORTED (5) JOB UPDATED (6)	<num>	0

	Meaning	Type	Default
<i>Type</i>	NO DATA (0), DATA BLOCK (1) DATA END (2)	<num>	0
<i>DataLength</i>	Number of bytes of image data in block	<num>	
<i>Width</i>	Width of scanned image in pixels	<num>	
<i>Length</i>	Length of the scanned image in pixels	<num>	
<i>JobId</i>	Job identification as defined by the host in the scan request	<num>	
<i>ErrorCode</i>	JOB NOT FOUND (0) RCF_INVALID_PARAM (2) RCF_MISSING_PARAM (3) RCF_SYNTAX_ERROR (4) MEMORY FULL (5) COMPRESSION OVERFLOW (6) CONTROLLER ERROR (7)	<num>	

---

Océ 9800 RCF/Job Ticket

---

*Programmer's Manual*

---

---

---

---

## Chapter 3

# Océ 9800 Repro Station

# Job Ticket Format specification

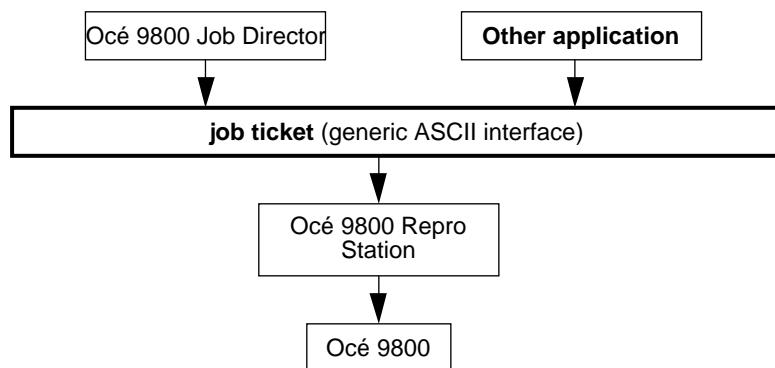
*The input format for Repro Station.*



# Introduction

**This chapter contains the specification of the Océ job ticket format as it is used in the current release of Océ Repro Station.**

A job ticket instructs the Océ machine how to handle a job. The specific settings can be made by a user on a control panel, or by means of the user interface (UI) on a workstation or PC. It does not really matter for the ticket specification how these settings are created. The user interface is independent from the ticket specification. The Océ job ticket acts as a generic ASCII interface.



Not everything you can specify in a ticket can be handled by a machine. You can specify duplex output in the ticket while the connected printers can only produce simplex output (e.g. Océ 6450). The printer device will select a fall back.

The purpose of a job ticket is to electronically send settings to a machine. The advantage of a general Océ job ticket is that you can send one job ticket to different machines and that all the machines understand the common functionality. For instance, specifying the number of copies is identical for all machines. A ticket specification leads to the same result on different machines if those machines support the functionality.

This chapter contains the specification of the Océ 9800 Repro Station Job Ticket.

‘General description’ on page 92 gives a global description of the format. The exact specification of the job ticket syntax is specified in ‘Detailed description’ on page 100.

All keywords are described in ‘Job Ticket keywords’ on page 108. An alphabetical list of all keywords can be found in the tables ‘structure keywords’ on page 129 and ‘setting keywords’ on page 129. ‘Implementation notes’ on page 131 gives some more details about the way applications should handle job tickets. The chapter concludes with examples of job tickets.

---

## General description

---

### What is a job ?

A job is defined as *any set of instructions, that belong together, resulting in the transformation of information from a source medium to a destination medium.*

Such a medium can be an electronic file, a paper document, or a microfilm card. The job settings describe how this transformation process has to be done.

IN \ OUT	file	paper	microfilm
file	copy	print	print
paper	scan	copy	scan + print
microfilm	microfilm scan	microfilm scan + print	copy

We distinguish the following types of job:

**print job** Electronic data is transferred to paper (including transparencies, polyester, vellum).

**copy job** Information on paper originals is transferred to paper.

**scan job (also called upload)** Information on paper originals is scanned and written to an electronic file (in some format).

**microfilm scan job** Microfilm (or aperture) cards are scanned and the result is written to an electronic file.

**mixed job** Electronic data is printed on paper, intermixed with copies of paper originals. It is even possible to mix electronic data and scanned data on the same page.

Scanning and printing of the scanned data can be combined in one job, these two steps make it possible to create a *copy job*. For example, copying microfilm information to paper or to another aperture card.

A job can process a single page, a number of pages, a document, or multiple documents. Currently in the “office” world an electronic document generally consists of one file containing multiple pages of text and drawings. For Engineering Systems, a job often consists of multiple documents of one page each.

The Océ Job Ticket contains all information necessary to control these various types of job.

**Release note:** *Océ 9800 Repro Station (current release) only supports print jobs.*

## What is a job ticket?

An Océ job ticket file specifies one job. In a job a number of *inputs* are processed and this results in a number of *outputs*. The job ticket specifies the sets of inputs, the sets of outputs and the processing that is needed to form the outputs.

An input can be a file, data appended after the ticket, a form, a paper page to be scanned, etcetera. In most cases the output is printed (print job), but it is also possible to store an output in a file (scan job).

Let us start with a simple example of a job ticket:

```
BeginTicket 1.0
  BeginOutput
    Name "info.tif"
    OutputSize A4
    Emulation TIFF
    Copies 3
  EndOutput
  BeginOutput
    Name "drawing.plt"
    OutputSize A0
    Emulation HPGL
  EndOutput
EndTicket
```

This ticket specifies 2 inputs and 2 outputs, but remains simple. All the settings that are not specified have *default values*. The number of copies of the second output is 1.

The Océ job ticket is used either as a separate file or it is placed before an existing document. A job ticket consists of a number of job ticket specification lines between a begin and an end line. If a document with a prepended job ticket is sent to a printer without any job ticket processing, the printer should handle the job ticket specification as comments: the printer should skip the job ticket. For this purpose the Océ job ticket uses a **token** at the start of each line.

---

**Release note:** *Océ 9800 Repro Station only supports job tickets that are separate files.*

## Token

The Océ job ticket starts with the line containing the string `BeginTicket`. This line and all the lines of the ticket start with a special token. A job from the Engineering Systems world may need a token that differs from a token used in the printing world and therefore the token string is *not fixed*. In principle, each Océ job ticket can have its own token string. However, one ticket has only one token. A ticket ends with a line containing a token, followed by the string `EndTicket`.

For recognizing and reading a job ticket, the token is not needed. Using a token is useful for easy recognition by a human reader or some other application. Another purpose of having a token is that you can specify all the ticket lines as comments for a certain Page Description Language (PDL). Different PDL's have different comment syntaxes and therefore having a variable token string is useful.

## Block structure

The Océ job ticket is block structured. There are several types of blocks: **output blocks**, **definition blocks** and **page blocks**. Simple job tickets only have one output block and no other blocks. When the processing described by the job ticket gets more complex, other blocks may be needed. There is one output block for each output.

## Output block

A ticket with only one output block:

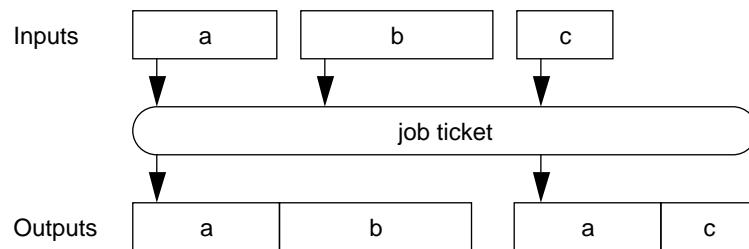
```
BeginTicket 1.0
  BeginOutput
    Directory c:\\ocers_q\\prfiles
    Name file.plt
    OutputSize A1
    MediaType paper
    Fold off
  EndOutput
EndTicket
```

The next example creates 3 copies of a polyester A3 unfolded and 1 copy of a plain paper A1 folded:

```
BeginTicket 1.0
  BeginOutput
    Directory c:\\ocers_q\\prfiles
    Name file1.plt
    OutputSize A3
    Fold off
    MediaType polyester
    Copies 3
  EndOutput
  BeginOutput
    OutputSize A1
    Directory c:\\ocers_q\\prfiles
    Name file2.plt
    Fold method standard
    +   size 210 297 mm
    +   edge 0 mm
    Emulation HPGL2
    Copies 1
    Mediatype paper
  EndOutput
EndTicket
```

## Definition block

In the next example we have 3 input files (called **a**, **b** and **c**) and we create 2 outputs. The first output consists of input files **a** and **b** (concatenated) and the second output consists of input files **a** and **c** (concatenated).



This is called a *matrix job*. Another way of looking at such a job is by showing the inputs and outputs in two dimensions. The outputs are delivered in so-called *sets*, with different settings for each set.

IN \ OUT	set 1 1 copy, A4	set 2 3 copies, A3
file a	X	X
file b	X	
file c		X

In the example another type of block is introduced: the definition block. A definition block is very similar to an output block. The main difference is that a definition block does not directly produce output, but its output is used by other blocks. Other blocks can reference definition blocks.

```

BeginTicket 1.0
  Directory "/usr/home/oce"
  Matrix
  BeginBlock 1
    Name "a"
  EndBlock
  BeginBlock 2
    Name "b"
    Emulation HPGL
  EndBlock
  BeginBlock 3
    Name "c"
  EndBlock
  Comment: Set 1
  BeginOutput
    OutputSize A4
    IncludeBlock 1 2
  EndOutput
  Comment: Set 2
  BeginOutput
    OutputSize A3
    Copies 3
    IncludeBlock 1 3
  EndOutput
EndTicket

```

The files "a", "b" and "c" are stored in the same directory. By specifying the directory outside the blocks it becomes a default for all the blocks. This default can be overruled by specifying the directory again inside a block.

The first 3 blocks are **definition blocks**. The keyword `BeginBlock` is followed by a name. Other blocks can reference definition blocks by this name. Output blocks generate output from input and this input can come from directly specifying an input file or by referencing a definition block.

In the output blocks the definition blocks are referenced by `IncludeBlock` followed by 2 names. The result is that the output of definition block 1 and the output of definition block 2 are concatenated and used as input in the output block.

In more complex matrix jobs, when each file in a set can have different settings, it is possible to define additional definition blocks (for each cell). In our example:

IN \ OUT	set 1	set 2
file a	1 copy A4	1 copy A3
file b	2 copies A0	
file c		3 copies A3

```

BeginTicket 1.0
  Directory "/usr/home/oce"
  Matrix
  BeginBlock file_a
    Name "a"
  EndBlock
  BeginBlock file_b
    Name "b"
    Emulation HPGL
  EndBlock
  BeginBlock file_c
    Name "c"
  EndBlock
  BeginBlock 1a
    Copies 1
    OutputSize A4
    IncludeBlock file_a
  EndBlock
  BeginBlock 1b
    Copies 2
    OutputSize A0
    IncludeBlock file_b
  EndBlock
  BeginBlock 2a
    Copies 1
    OutputSize A3
    IncludeBlock file_a
  EndBlock
  BeginBlock 2c
    Copies 3
    OutputSize A3
    IncludeBlock file_c
  EndBlock
  Comment: Set 1
  BeginOutput
    IncludeBlock 1a 1b
  EndOutput
  Comment: Set 2
  BeginOutput
    IncludeBlock 2a 2c
  EndOutput
EndTicket

```

---

**Release note:** Océ 9800 Repro Station (current release) does not allow nesting of *BeginBlocks* (as in the last example above).

---

## Detailed description

---

### Syntax

**line oriented** Each Océ job ticket setting is specified on a separate line. A line is terminated by a carriage return character (CR, octal 015), a newline character (LF, octal 012), or a combination (CR/LF or LF/CR). The maximum ticket line length is 255 characters (including the line termination characters).

**empty lines** Empty lines inside a ticket are allowed.

**continuation lines** If the maximum ticket line length is not enough to specify a setting, a continuation line can be used. A continuation line starts with the token, followed by the + sign.

```
<token>Notes "This is a very long"  
<token>+ " first notes line,\nfollowed by"  
<token>+ " a second line."
```

In the example above, notes consists of 2 lines, which takes up 3 lines in the ticket. The continuation line can be used without actually reaching the maximum line length. This is useful for readability. In the example it would make more sense to split the lines at the newline character. But the example shows that newline characters have to be inserted on purpose. The physical end-of-line does not automatically add a newline character.

**keywords / values** The job ticket settings consist of keywords and values. Values consist of strings, integers, reals or also keywords. Examples:

```
Name "report.ps"  
Zoom 220 auto
```

The first setting has one value, a string. The second setting has 2 parameters, a real value and a keyword.

**unknown keywords** Lines that contain unknown keywords are ignored. Also if a value is not correct, this setting is ignored.

**order of keywords** The order of keywords is not important, except when the same keyword is used multiple times, or when it appears at another block level.

**multiple keywords** If a keyword appears multiple times at the same level, the first occurrence is used. So the first one wins. If the keyword appears at a "deeper" level, the first one at that level overrules the previous setting. This allows defaults for blocks or for pages to be set at job level.

**whitespace** Keywords and values are separated by whitespace (any combination of space (octal 040) and horizontal tab (octal 011) characters).

**case sensitivity** All keywords are case-insensitive.

The token is case-sensitive.

**7 bits ASCII** All ticket lines contain 7-bits ASCII. For specification of 8-bit characters (e.g. accented characters inside a string) the ISO Latin 1 encoding is used. Those characters are specified by their octal value, see below.

Example:

```
Name Oc\351
```

**string specification** Strings, e.g. filenames, can be written between double quotes or not. The use of double quotes is not needed if there is no whitespace inside the string. When a string contains a double quote, this double quote is escaped by a backslash. The backslash itself is also escaped by a backslash.  
Examples:

```
Name "report.ps"
Name report.ps
Notes "Oc\351\tRepro Station\n"
Notes "Backslash \\ and double quote \" inside a string\n"
```

**backslash** A backslash is used for specifying 8-bit characters and also for specifying a double quote or a newline character inside a string. A backslash has a special meaning if it is followed by:

- double quote (double quote inside string)
- backslash (backslash inside string)
- n (newline inside string)
- r (carriage return inside string)
- t (horizontal tab inside string)
- 3 octal digits (8-bit ISO Latin 1 ASCII character inside string)

The \ddd form may be used to include any 8-bit character constant in a string. One, two, or three octal digits may be specified with high-order overflow

ignored. This means that for all characters above \377, \400 is subtracted, to yield a result that is at most 255 decimal. Examples:

- "\a" the backslash is ignored, string "a"
- "\\\b" backslash escaped, string "\b"
- "\\\n" backslash escaped, string "\n" (no newline)
- "Oc\351" string "Océ"
- "\70abc" string "8abc" (\070 is "8")
- "\470abc" string "8abc" (\470 becomes \070)
- "\709abc" string "89abc" (9 is not octal)

This notation is required for specifying a character outside the supported 7-bit ASCII character set for the job ticket contents. The notation itself stays within the standard set and thereby avoids possible problems in transmitting, storing or editing the job ticket.

**number specification** Numerical values can be integers or reals. E.g. page numbers are specified as integers, a scale factor is specified either as an integer or as a real. For some values keywords can be specified. Unit specification is needed whenever a number is used as a measurement value. Possible unit values are:

- cm centimetre value
- mm millimetre value
- inch inch value
- pt point (1/72 inch) value

Examples:

```
Zoom 200.0 200
Zoom 200 auto
Shift 2 -3.4 inch
Pens number 1-9 width 0.15 mm
```

---

## Token

**preceding whitespace** On some systems, characters on a specific position on a line have special meaning. The Océ job ticket allows variable whitespace (spaces and/or tabs) before the token. This whitespace can also be used for indentation to improve the readability of the ticket.

An Océ application determines the token by scanning the first line of the ticket. This line should contain the string `BeginTicket`. If the job does not start with this line, then the job does not contain a job ticket. The characters on the first line before the string `BeginTicket` form the token, except any preceding whitespace. The job ticket ends with the line `<token>EndTicket`.

**embedded whitespace** The token itself can contain embedded or trailing whitespace (spaces and/or tabs).

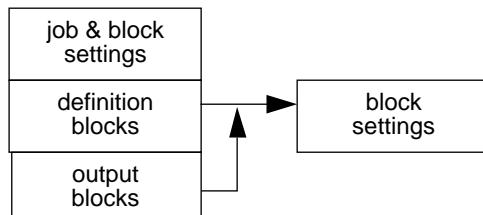
Example of a job ticket for a Océ 9800 environment with token `%!Oce` and some extra preceding whitespace. There is an empty line after the first block.

```
%!OceBeginTicket 1.0
  %!OceBeginOutput
    %!OceName "drawing.plt"
    %!OceCopies 3
  %!OceEndOutput

  %!OceBeginOutput
    %!OceName "banner.txt"
  %!OceEndOutput
%!OceEndTicket
```

## Job ticket layout

In the job ticket structure you must first specify the settings for the job and the default settings for all blocks. Then all definition blocks are specified and finally all output blocks.



If settings appear at job level, but after a block has been defined, they are ignored. The order of blocks is not restricted.

---

## Block structure

**reference levels** In the example in the general description, there is one level of referencing. However, within a definition block there can be a reference to another definition block, increasing the level of references. The Océ ticket specification does not restrict the number of levels. Circular referencing is not allowed. With this block referencing scheme we can freely combine inputs and outputs and also can handle complicated processing.

When we reference a block, we use the output of that block as input in the current block. The current block itself can process its input and feed another block. This mechanism can be compared with the *UNIX pipe mechanism*, output from one program is the input for the next program. For complex operations you may need several pipes. A difference is that the Océ job tickets can handle several inputs and several outputs.

A ticket can have one or more reference levels, but there is **only one block level**: there is no `BeginBlock` between a `BeginBlock` and `EndBlock` statement (it is ignored).

**Release note:** *Océ 9800 Repro Station (current release) supports one level of referencing only.*

---

## Input/output handling

Each block in a ticket has one or more inputs and one output. Output of a definition block becomes input of the block that references the definition block. Output of an output block is the final output. A job has more than one output if there is more than one output block.

The job ticket describes the final output after a number of processing steps by the blocks. There are several ways to specify block input:

**indirect input** By referencing a definition block, the output of that block becomes the input of the current block. This is called indirect input.

**direct input** If an input is not a reference to a definition block, the input is appended after the ticket or is an external source. Such an input reference is called "direct".

Both direct and indirect input can be used in keywords like `IncludeBlock`.

## Direct input

**type of input** The keyword `Type` specifies which kind of medium the input or output is. It can be either `file`, `paper`, or `microfilm`. Default for input is `file`, default for output is `paper`. So this keyword must be used when scan jobs or mixed jobs are specified.

A block can specify file input by using the `Directory` and `Name` keywords. `Directory` specifies the input directory and `Name` the input filename. It is allowed to specify the complete path under `Name`. For input types other than `file`, the `Name` identifies another source, e.g. a paper original.

```
BeginOutput
  Name "drawing 0823415"
EndOutput
```

**default: atend** If no input is specified in a block, the input data starts after the ticket specification in the file. This can also be specified explicitly:

```
IncludeBlock atend
```

Keyword `atend` is reserved, it may not be used as identifying string for a definition block.

**Release note:** *`atend` is not supported by Océ 9800 Repro Station (current release). Furthermore, it handles only jobs that have `file` input and `paper` output.*

## Indirect input

Indirect input is specified by referencing other blocks. The input is primarily specified by `IncludeBlock`. This primary input may be overlayed or underlaid by other input, using `OverlayBlock` and `UnderlayBlock`.

**IncludeBlock** Specifies indirectly where the input comes from. If more than one input is specified, the inputs are concatenated.

```
IncludeBlock one two atend "examples.ps"
```

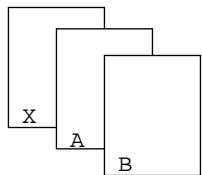
The example shows 4 concatenated inputs. The first and second input reference a definition block. These are indirect inputs. `atend` specifies that the third input is appended after the ticket, and the fourth input is an external file. These two are direct inputs.

---

**Release note:** `atend` and referencing a file directly are not supported in Océ 9800 Repro Station (current release).

**OverlayBlock** Specifies which input is overlaid on top of the primary input, as identified by Name or `IncludeBlock`. If more than one input is specified, the inputs are all overlaid. Each overlay comes on top of any preceding overlay, on top of the primary input.

```
IncludeBlock X
OverlayBlock A B
```

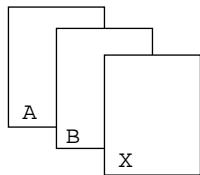


---

**Release note:** Overlays are not supported by Océ 9800 Repro Station (current release).

**UnderlayBlock** Specifies the inputs that are used as underlays. If more than one input is specified, the inputs are all underlaid. Each underlay comes on top of any preceding underlay, below the primary input.

```
IncludeBlock X
UnderlayBlock A B
```



---

**Release note:** *Underlays are not supported by Océ 9800 Repro Station (current release).*

## Order of processing

Page contents can be shifted, scaled, rotated, and so on. The relevant settings are:

- CalComp, HPGL, HPGL/2, ASCII
- LongPlot
- RemoveStrip
- Zoom
- Rotate
- OutputSize
- Stamp
- Shift
- AddStrip

Because the order in which settings are executed influences the result, these settings will be handled in a fixed order. However, the job ticket keywords can be specified in any order, as described before. The order is as specified in the preceding list. So rotation is done after scaling.

---

## Job Ticket keywords

In this section all job ticket keywords and their values are specified. The keywords are presented in two categories: structure keywords and setting keywords. Structure keywords are related to the logical (block) structure of the job. Setting keywords specify processing steps for the device.

The keywords are listed in alphabetical order.

The syntax is specified in Backus-Naur form (BNF):

```
<clause>      Indicates a clause item. It may be an elementary
               type or comprise other clause
 ::=           "is defined as"
 [ expression ] the expression inside the brackets is optional
 { expression } groups expressions or clause into a single
               expression
 <clause> ...  ellipsis, one or more instances of <clause> can
               be specified
 |
 <string> ::=  string of characters, may be enclosed by double
               quotes
 <int> ::=    integer, consists of any combination of the
               characters 0, 1, 2, 3, ..., 9
 <char> ::=   <int> in range [0..255]
 <real> ::=   real number (includes <int>),
               in the format
               <int>.<int> or <int>.<int>E<int>.
```

# Structure keywords

## BeginBlock

Indicates the start of a definition block. The block is identified by the string and can be referenced in an `IncludeBlock`'s value. Note that when a definition block with the same name is specified, the contents of the second block are ignored.

The `BeginBlock` keyword must have a matching `EndBlock`.

**syntax** `BeginBlock <string>`

## BeginOutput

Indicates the start of an output block. The `BeginOutput` keyword must have a matching `EndOutput`.

**syntax** `BeginOutput`

## BeginTicket

Indicates the start of the Océ job ticket. Lines before the line with this keyword are ignored. The characters preceding `BeginTicket` (except leading white space) are interpreted as the **token**.

Only lines with this token are recognized to contain valid job ticket information. The token is case-sensitive. The `BeginTicket` keyword must have a matching `EndTicket` keyword.

The `BeginTicket` keyword is followed by a version number. The version number of the ticket described in this document is 1.1

**syntax** `BeginTicket <version number>`

**example** `%!OceBeginTicket 1.1`

## **EndBlock**

This keyword terminates a definition block. It does not have a value.

**syntax** `EndBlock`

## **EndOutput**

This keyword terminates an output block. It does not have a value.

**syntax** `EndOutput`

## **EndTicket**

Indicates the end of the Océ job ticket. It does not have a value. The first character on the line after `EndTicket` is possibly the start of data that is appended to the job ticket. This input can be referenced by using the `atend` reserved value in combination with the `Name` or `IncludeBlock` keywords.

**syntax** `EndTicket`

## **IncludeBlock**

Specifies the inputs in a block. The order in which the values appear, determines the order in which the inputs are processed. All the included blocks are **concatenated**.

**syntax** `IncludeBlock { <string> | atend }...`

`IncludeBlock` is followed by one or more strings. A string can refer to a definition block, or, when the definition block is not found, to input appended after the ticket or to external input (e.g. a file). The reserved word `atend` indicates that the input is appended to the ticket.

**example** `IncludeBlock one "report.ps" atend`

The example includes 3 inputs: the first input is the output from a definition block with name "one", the second input is a file with name "report.ps" and the third input follows the ticket specification.

**default** The default is atend. If no IncludeBlock is specified the input is expected to follow the ticket.

---

**Release note:** *atend and referencing a file directly are not supported by Océ 9800 Repro Station (current release).*

# Setting keywords

## Account

The string value indicates the account that can be billed. This keyword can occur at job level where it indicates who has to be charged for processing this job. It can also appear in an output block so the recipient of the output is asked to pay for it.

Furthermore, it is possible to attach the account to a definition block where it can be used to charge for the retrieval costs of the input (e.g. retrieving a paper drawing from the archive).

Account consists of some free text lines. The mapping of these text lines is client specific. A client can specify that the first line is the name, the second is the address and the third is the city. Or the first line is the department number and the second line is a project number. The lines are separated in the string by a newline character.

**syntax** Account <string>  
**default** Not applicable.

**example** Account "Oc\351-Nederland B.V.\nP.O. Box 101\n"  
+ "5900 MA Venlo"

**Release note:** For Océ 9800 Repro Station (current release) only used at job level.

## AddStrip

This keyword specifies how much whitespace must be added to the paper output at the leading edge or the trailing edge. The use of the unit is mandatory. Note that this keyword can appear in combination with RemoveStrip; they are not exchangeable.

**syntax** AddStrip <addstrip> ...  
<addstrip> ::= { leading <real> <unit> } |  
{ trailing <real> <unit> }  
**default** 0

---

## ASCII

The **ASCII** keyword specifies how ASCII files are handled. The value consists of a number of optional parameters each followed by one or more settings. If no parameters are present, this keyword has no effect.

The **line** parameter indicates whether a line that extends beyond the right margin is wrapped into the next line, or is truncated. The end of line (**eol**) parameter specifies which character(s) indicates the end of a line.

The parameters **top**, **bottom**, **left**, and **right** specify the margin along the edge of the page where no characters are printed. The use of the unit is mandatory for each parameter.

The **fontsize** specifies the size of the used font in points. The unit is fixed and must be **pt** (1 point = 1/72 inch).

Currently no font type can be selected; the device determines which font is used. Normally this is a Courier-like font. Note that ASCII settings can be specified even when the job does not contain any ASCII inputs or outputs.

```
syntax ASCII <ascii> ...
<ascii> ::= { line { wrap | trunc } } |
           { eol { cr | crlf | lf } } |
           { top <real> <unit> } |
           { bottom <real> <unit> } |
           { left <real> <unit> } |
           { right <real> <unit> } |
           { orientation { portrait | landscape } } |
           { fontsize <real> pt }
```

**default** None

---

## CalComp

The **CalComp** keyword specifies how CalComp files are handled. The value consists of a number of optional parameters each followed by one or more settings. If no parameters are given, this keyword has no effect.

The **origin** determines the location of the reference point for all coordinates that are used in the CalComp file.

The `checksum` specifies whether the checksum is on or off.

The parameter `doublesync` indicates whether a single synchronisation character (off) is used or two(on). Which character is the synchronisation character is determined by the `sync` parameter.

The `stepsize` determines in which "resolution" or accuracy the output must be generated. It is always expressed in steps per inch. The end of message (`eom`) parameter specifies which character terminates the drawing.

Currently it is not possible to specify separate pen settings for CalComp. The keyword `Pens` is used for all vector file formats.

Note that CalComp settings can be specified even when the job does not contain any CalComp inputs.

```
syntax CalComp <calcomp> ...
<calcomp> ::= { origin { ul | upperleft | ur | upperright | ll
    | lowerleft | lr | lowerright | center } } |
    { checksum { on | off } } |
    { doublesync { on | off } } |
    { stepsize <int> } |
    { eom <char> } |
    { sync <char> }
```

**default** None

---

## Collate

Specifies the sorting method of an output block's result when multiple copies are requested (see Copies). Setting this value 'on' means that the copies are sorted by set (for example: 1-2-3-1-2-3-1-2-3), 'off' indicates that multiple copies are sorted by page (1-1-1-2-2-2-3-3-3).

```
syntax Collate { on | off }
default on
```

---

## Comment

This keyword is the normal way of adding remarks to the job ticket. They can be used to clarify certain structures or settings and make the contents better understandable. This keyword has no influence on the processing of the ticket.

**syntax** Comment <any>  
**default** Not applicable.

---

## Copies

Indicates the number of copies that will be generated of an output block. The number must be larger than zero. If the output is a file (for a scan job), it is ignored and defaults to 1. For an input it is also ignored.

**syntax** Copies <int>  
**default** 1.

In the current version of Repro Station, the copies keyword is used slightly different for a normal job and a matrix job.

In a normal job, the Copies keyword inside an output block is ignored. Only the (default) copy settings specified at ‘job’ level is used. This makes sense because you can only specify the number of copies for the complete job. In a matrix job, the Copies keyword inside an output block overrules the default Copies and it specifies how many outputs of each set must be made.

---

## CopyMatrix

The copy matrix is an array that maps all supported paper sizes to another (or the same) paper size. The original size of the image is the first part of each parameter, the destination size is the second part. They are separated by a colon (octal 072).

This setting only has effect when size is either ‘auto’ or not explicitly set to a paper size, and zoom is set to ‘auto’.

The different paper sizes fall into 5 categories. This means that not all paper sizes can be intermixed. The standard rule is that the first setting wins also applies to the original to copy matrix. In other words, if you specify an original paper size more than once the first one is valid. So if

`"CopyMatrix A0:A1 E:B"`

is specified, an E-sized original will be printed on D (same category as A1).

The categories are:

- 1** A0, 34"(E), 36"(E+), 700 mm, 30"
- 2** A1, 22"(D), 24"(D+), 500 mm
- 3** A2, 17"(C), 18"(C+)
- 4** A3, 11"(B), 12"(B+)
- 5** A4, 8.5"(A), 9"(A+), letter, legal

For example, when you want all A0 and A1 sizes to be printed on A1, and smaller formats on the correct size, you specify

```
CopyMatrix A0:A1 A1:A1 A2:A2 A3:A3 A4:A4
syntax CopyMatrix <orgtocopy> ...
<orgtocopy> ::= <papersize>:<papersize>
default A one-to-one mapping of paper sizes (A0:A0, A1:A1, A2:A2, A3:A3, A4:A4)
```

## Copyright

The value is an arbitrary string indicating the origin of the job ticket. This keyword does not affect the processing of the ticket.

```
syntax Copyright <string>
default Not applicable.
```

## CreationAppl

The value is an arbitrary string indicating which application generated the job.

```
syntax CreationAppl <string>
default Not applicable.
```

## CutMethod

The value indicates whether the paper output must be cut at standard size, at the size of the image (synchro), or at a specific length.

Note that when used in combination with AddStrip, the edges are added last, so standard does not necessarily mean that the output is a standard size.

```
syntax
CutMethod { standard | synchro | { <real> <units> } }
default synchro
```

## Directory

This keyword specifies the directory where a file can be found, or must be written if it applies to an output.

The path should be accessible from the system where the job ticket is processed. Both the "\\" and "/" characters can be used to separate directories; they are interpreted by the application that processes the job ticket.

**syntax** Directory <string>  
**default** Device dependent.

## Distribution

The value contains information that is important for the distribution of the output that is produced by the job. It may contain a list of addressees to whom copies of the output must be sent, or information about off-line finishing requirements. This information may be printed on a banner page.

This keyword can appear at job-level when it contains information about the job as a whole. It can also be used in an output block, where it only says something about the output of this block.

**syntax** Distribution <string>  
**default** Not applicable

## Emulation

This keyword determines the format of a file. When `auto` is chosen, the device will decide what format the file is.

**syntax** Emulation { `auto` | `ASCII` | `CalComp` | `CALS` |  
`HPGL` | `HPGL/2` | `HPRTL` | `TIFF` }

**default** Auto.

---

## Fold

This keyword specifies whether the output has to be folded.

Off means that output is not folded. On means that folding is done in a standard way determined by the device.

If folding must be done, but not in the standard way, at least one of the "custom" parameters must be specified. The method parameter selects the type of folding which results in the legenda on top. The size parameter must be followed by the foldwidth, then the foldlength, then the unit. Both values must be expressed in the same unit. Orientation specifies how the page must be oriented before it is folded. The width of the binding edge is specified by the edge parameter.

### **syntax**

```
Fold      off | on | <customfold>
<customfold> ::= <foldoption> ...
<foldoption> ::= { method { standard | ERICSSON | AFNOR } } |
{ size <width> <length> <unit> } |
{ orientation { auto | portrait } } |
{ edge <real> <unit> }
<width> ::= <real>
<length> ::= <real>
```

**default** off.

---

## HPGL

The HPGL and HPGL/2 keywords specify how HPGL respectively HPGL/2 files are handled. The value consists of a number of optional parameters each followed by a setting. If no parameters are given, this keyword has no effect.

The **plottertype** specifies for which plotter type the file was generated. For a valid list of plotter types see figure 28 on page 61 and see figure 31 on page 63. The **origin** determines the location of the reference point for all coordinates that are used in the file. Parameter **sp\_eof** specifies whether selection of pen 0 in the file must be treated as an end-of-file (on) or not (off).

Currently it is not possible to specify separate pen settings for HPGL or HPGL/2. The keyword **Pens** is used for all vector file formats.

Note that HPGL and HPGL/2 settings can be specified even when the job does not contain any HPGL or HPGL/2 inputs or outputs.

```
syntax HPGL    <hpgoption> ...
<hpgoption> ::= { plottertype <string> } |
                 { origin { ul | upperleft |
                            ur | upperright |
                            ll | lowerleft |
                            lr | lowerright |
                            center } } |
                 { sp_eof { on | off } }
```

**default** None.

## HPGL/2

See the HPGL keyword.

## JobCollate

Setting this value on means that the output is sorted by set. The copies are produced setwise (for example, 1-2-3-1-2-3-1-2-3). Off indicates that multiple copies are sorted by page (1-1-1-2-2-2-3-3-3).

```
syntax JobCollate { on | off }
default on
```

## JobFlagsheet

Flagsheets are banners for each job or set. Flagsheets offer the possibility to display the contents of identification job attributes.

When multiple copies are printed all sets of the job are provided with a flagsheet.

```
syntax
JobFlagsheet      <flagoption>...
<flagoption> ::= { text <string> } | <flagpos>
                  { tray <int> } | { size <papersize> }
<flagpos> ::= header | trailer
```

**default** No Flagsheets at front or end of the job. Default position is header.

**Release note:** *Océ 9800 Repro Station (current release) does not support text, flagpos (always header) and tray. You can specify size.*

## JobName

The name of the job.

**syntax** JobName <string>

**default** Not applicable.

## Matrix

This keyword indicates that the job is a matrix job. This means that output blocks have to be interpreted as sets. When not present, the ticket is interpreted as an ‘ordinary’ job.

**syntax** Matrix  
**default** no matrix job

---

MediaFeed

This keyword selects a fallback strategy when it is not possible to print on the specified paper size.

**Larger** indicates that it can be printed on the specified paper size or on a larger format if available, but not on a smaller format.

**Smaller** means that the document can be printed on the specified paper size or on a smaller size, but not on a larger one. If a smaller size is chosen the image will be scaled to fit on the paper.

Any means that the document can be printed on any paper size. The printer selects the exact size first, next a larger size, and finally a smaller size. If a smaller size is chosen the image will be scaled to fit on the paper.

Exact means that printing must be done on the specified paper size. If this is not available on the device, the job cannot be processed until the operator has installed the correct paper size.

When manual is selected, the operator must feed paper manually into the device. Note that still the OutputSize can be specified, so the operator is informed about the paper size that is required.

**syntax** MediaFeed { larger | smaller | any | exact | manual }  
**default** exact

---

**Release note:** *Océ 9800 Repro Station (current release) does not support the keyword 'smaller'.*

---

## MediaType

This keyword specifies the media the document has to be printed on.

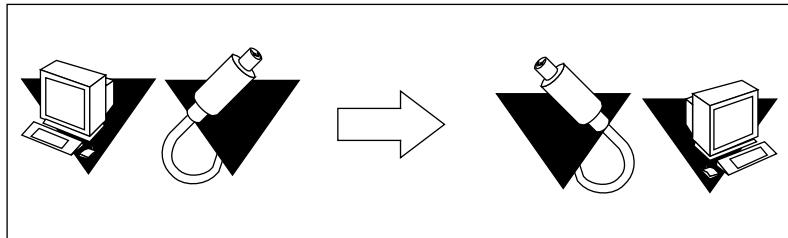
**syntax** MediaType { paper | transparent | polyester }  
**default** paper

---

## Mirror

This keyword specifies whether the image must be mirrored along the vertical axis.

**syntax** Mirror {on | off}  
**default** Off



[38] Selecting mirror

## Name

The name of an item. For a file this is the filename (path not included). For a paper original, the name specifies which original this is. If an application spools a file, this setting contains the name of the spooled file.

**syntax** Name <string>  
**default** Not applicable

## Notes

Notes consists of some free text lines. These lines can be used for additional remarks for the recipient of the job ticket, usually an operator.

**syntax** Notes <string>  
**default** Not applicable.

## OrigDirectory

This keyword identifies the absolute path where the original file can be found. This is the directory where the file was located when it first was included in the job.

Note that when a job is sent or copied, the print files may be copied as well, so the original directory and filename do not point to the file to be used.

**syntax** OrigDirectory <string>  
**default** Not applicable

---

## OrigName

This keyword identifies the original file's name.

**syntax** OrigName <string>  
**default** Not applicable

---

## OutputSize

This keyword specifies the paper size that will be used for the output. If this size is not available, the **MediaFeed** keyword specifies if a fallback can be used. The value **auto** means that paper should be used that matches the image's size after stripping (**RemoveStrip**) and scaling (**Zoom**).

```
syntax OutputSize { auto | <papersize> }
<papersize> ::= { a0 | a1 | a2 | a3 | a4 | a5 |
                    A | B | C | D | E |
                    A+ | B+ | C+ | D+ | E+ | 30inch |
                    700mm | 500mm | letter | legal | ledger }
```

**default** auto

**Release note:** *Repro Station 9800 (current release) does not support letter, legal, ledger, or a5.*

---

## Pens

This keyword specifies for all vector print files which pens must be used. When **off** is selected, the pen settings as specified in the file itself are used. When **on** is selected, the pen settings as defined on the device will be active.

It is possible to specify "custom" settings by specifying for one or more pens the pen width and the pattern. The pen numbers range from 1 to 999. The **width** and **pattern** parameters apply to the previously defined pen number(s). When the **width** is specified, the unit is mandatory. The **pattern** is an enumeration with 1 = white, 16 = black, intermediate values indicate a gray value. Numbers 17 through 32 are used for special patterns.

The following example sets pen 10 to a width of 1.1 mm, pens 11 up to and including 16 to 0.5 mm and all other pens to 1 mm. All pens have pattern 16 (black), except pen 12 which is gray.

```

Pens number 1-9 width 1 mm pattern 16
+      number 10 width 1.1 mm pattern 16
+      number 11 width 0.5 mm pattern 16
+      number 12 width 0.5 mm pattern 7
+      number 13-16 width 0.5 mm pattern 16
+      number 17-999 width 1 mm pattern 16
syntax Pens { off | on | <custompens> ... }
<custompens> ::= { number { <int> | <int>-<int> | all}
                  <pensetting> ... }
<pensetting> ::= { width <real> <unit> } |
                  { pattern <int> }
default off

```

## Punch

The **Punch** keyword specifies whether the output must be punched or not. If **on** is selected, punching is done in a way determined by the device.

```

syntax Punch { off | on }
default off

```

## RemoveStrip

This keyword specifies how much information must be stripped from the image. You can specify separate values for the leading and trailing edges. The **unit** is mandatory. This setting only applies to inputs.

```

syntax RemoveStrip <removestrip> ...
<removestrip> ::= { leading <real> <unit> } |
                  { trailing <real> <unit> }
default Nothing is stripped.

```

## Rotate

This keyword specifies the angle by which the input must be put on the output (usually paper). The angle is given in degrees and is measured counter-clockwise.

It is possible to specify "custom" settings by specifying four different rotation angles. If you specify **portrait** the image will be rotated to portrait

orientation. This is used often to fold originals with a legend. If you specify `landscape`, the orientation of the image will be changed to landscape if possible. Images in landscape orientation can be printed faster because this requires less paper movement through the machine. `Portrait180` and `landscape180` both add an extra rotation of 180 degrees to place the legend in the right corner.

**Note:** *An original in landscape orientation has its shortest side directed parallel to the direction of the paper movement. An original in portrait orientation has its longest side directed parallel to the paper movement.*

**syntax** `Rotate { 0 | 90 | 180 | 270 | portrait | landscape | portrait180 | landscape180 }`  
**default** Device dependent.

---

## Shift

This keyword specifies that the image must be shifted in the x or y direction. When two numbers are given, the first indicates the x-shift (perpendicular to the paper path) and the second the y-shift (direction of the paper path, positive number is "up"). The unit is mandatory.

It is also possible to shift to one of the corners of the page (autoshift).

**Note:** *Custom Shift and Auto Shift can be specified together: The automatic shift will be done first, followed by the custom shift.*

**syntax**  
`Shift { <customshift> | <autoshift> }...`

`<customshift> ::= <real> <real> <unit>`  
`<autoshift> ::= ul | upperleft | ur | upperright |`  
`ll | lowerleft | lr | lowerright |`  
`ce | center`

**default** No shift.

---

## Stamp

This keyword selects the stamp mechanism of the device. If the setting is `off`, no stamp is printed on the output. If `on` is selected, a stamp is put on each output page or range of pages. The position, size, and contents of this stamp

are determined by the key operator of the Océ 9800 printer. If selection of a "custom" stamp is required, at least the stamp number or text must be specified. The other parameters determine where the stamp is placed, what size it is, and what color.

**syntax**

```
Stamp           { off | on | <customstamp> }
<customstamp> ::= <stampid> [ <stampoption>... ]
<stampid> ::= { number <int> } | { text <string> }
<stampoption> ::= { position { top | middle | bottom } } |
                  { size { small | large } } |
                  { color { black | darkgray | lightgray |
                            gray } }
                  }
```

**default** off

---

**Release note:** Océ 9800 Repro Station (current release) does not support the text keyword.

---

## UserName

The name of the user.

**syntax** UserName <string>  
**default** Not applicable.

---

## Zoom

This keyword specifies the zoom factor for the image. The scaling in x direction is specified first. Optionally the scaling in y-direction can be given. By default the x-scaling is used in the y-direction. A scaling value of 100 means a scaling of 100 %. A value of 0 is not allowed, in that case it defaults to 100. Specifying auto means that the image is scaled to fit the output paper size.

If Zoom auto is specified, then the scaling in the y-direction will be the same as in the x-direction. So scaling in both directions is the same. Zoom auto auto may result in different x and y-scaling factors.

```
syntax
Zoom { auto | <x> } [ auto | <y> ]
  <x> ::= <real>
  <y> ::= <real>
default 100 100
```

---

**Release note:** *Océ 9800 Repro Station (current release) does not support anamorphic zoom, so scaling in X and Y direction must be the same.*

---

## Overview

Table 39 lists all structure keywords that are recognized by Océ 9800 Repro Station (current release) and indicates at which level they can be used.

keyword	job level	def. block	output block
<i>BeginBlock</i>	*		
<i>BeginOutput</i>	*		
<i>BeginTicket</i>	*		
<i>EndBlock</i>	*		
<i>EndOutput</i>	*		
<i>EndTicket</i>	*		
<i>IncludeBlock</i>			*
[39] structure keywords			

A \* indicates that the keyword can occur at this level.

Table 40 lists all settings keywords that are recognized by Océ 9800 Repro Station (current release) and shows what their meaning is when they are used at job level. A \* indicates that the setting applies to the job as a whole, *def* means that it is used as a default value for inputs and/or outputs.

The last two columns indicate whether the setting applies to inputs or outputs. If a \* is shown for input and output then the setting has a meaning for both. If nothing is shown, the setting is ignored for this particular type.

setting	job level	input	output
<i>Account</i>	*		
<i>AddStrip</i>	def		*
<i>ASCII</i>	def	*	
<i>CalComp</i>	def	*	
<i>Comment</i>	*	*	*
<i>Copies</i>	def		*
<i>CopyMatrix</i>	def		*
[40] setting keywords			

<b>setting</b>	<i>job level</i>	<i>input</i>	<i>output</i>
<i>Copyright</i>	*		
<i>CreationAppl</i>	*		
<i>CutMethod</i>	def		*
<i>Directory</i>	def	*	
<i>Distribution</i>	*		
<i>Emulation</i>	def	*	
<i>Fold</i>	def		*
<i>HPGL</i>	def	*	
<i>HPGL/2</i>	def	*	
<i>JobCollate</i>	*		
<i>JobFlagsheet</i>	*		
<i>JobName</i>	*		
<i>Matrix</i>	*		
<i>MediaFeed</i>	def		*
<i>MediaType</i>	def		*
<i>Mirror</i>	def	*	
<i>Name</i>	def	*	
<i>Notes</i>	*		
<i>OrigDirectory</i>	def	*	
<i>OrigName</i>	def	*	
<i>OutputSize</i>	def		*
<i>Pens</i>	def	*	
<i>Punch</i>	def		*
<i>RemoveStrip</i>	def	*	
<i>Rotate</i>	def	*	
<i>Shift</i>	def	*	
<i>Stamp</i>	def		*
<i>UserName</i>	*		
<i>Zoom</i>	def	*	

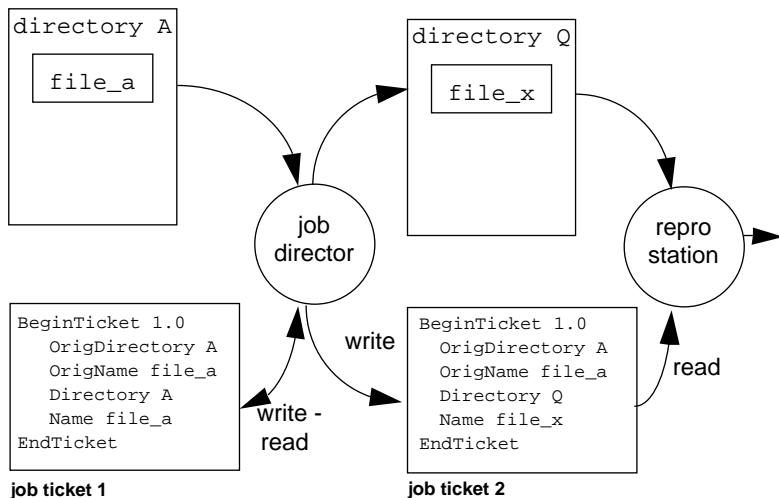
[40] setting keywords

# Implementation notes

This chapter contains guidelines for handling job tickets.

## Handling job tickets

**writing job tickets** Because the job ticket can be a separate file that references input files, it is sometimes difficult to move a jobticket without losing the link to the referenced files. Only when the information is appended to the ticket the coupling is straightforward (not supported by the current release of Océ 9800 Repro Station). For this reason, the Name/Directory and OrigName/OrigDirectory are used.



The diagram above shows two applications, Job Director and Repro Station. Job Director creates a job ticket which contains a reference to file\_a in directory A.

If this job ticket is *saved* (job ticket 1), both OrigDirectory and Directory will be "A". Also, both OrigName and Name will be "file\_a".

If Job Director *sends* the job to Repro Station, a new job ticket will be created with OrigDirectory "A" and OrigName "file\_a" (job ticket 2). Job Director also copies file\_a to directory Q and the file will be called "file\_x". Therefore, Directory will be "Q" and Name will be "file\_x".

If directory Q is not specified in the jobticket, Repro Station uses a default directory (PRFILES). When files are written across a network into a directory, it is advised to **not** specify the Directory keyword. The two systems may refer to the same directory with a different name.

For example: Job Director on PC 1 writes print files to directory **h:\inqueue**, and Repro Station on PC 2 reads the data from directory **g:\inqueue**. Directory can only be used when it is absolutely sure that the recipient refers to the directory in exactly the same way.

Note that an application that generates a job ticket and copies the print files to a queue directory must make sure that no existing files are overwritten. Make sure that print files are present in the queue directory, before the job ticket is written there.

**deleting job tickets** When a job ticket has to be deleted, the application must know if it can also remove the referenced files. The rule is that if the original name (plus directory) is equal to the name, the file may not be deleted. But if these names differ, the file has been copied especially for the job ticket, so can be removed.

---

## Typical examples for Océ 9800 Repro Station

---

### Example 1 Normal job

```
%%OCE%%BeginTicket 1.1
%%OCE%%Copyright 1995 Oce-Nederland B.V.
%%OCE%%CreationAppl      "Oce 9800 Repro Station R1.1"

%%OCE%%Comment          Job Settings
%%OCE%%JobName          "Birdie"
%%OCE%%UserName         "Charlie Parker"
%%OCE%%Account          "Ornithology"
%%OCE%%Notes            "print this job before 4 PM"
%%OCE%%Distribution     "1 set to Mr Jacobs\r\n1 set to "
%%OCE%%+                "Ms Kaye\r\n1 set to Central Archive\r\n"
%%OCE%%JobFlagSheet     size A4
%%OCE%%Copies            3
%%OCE%%JobCollate       on
%%OCE%%CopyMatrix       A0:A0 A1:A1 A2:A2 A3:A3 A4:A4

%%OCE%%Comment          Input Defaults
%%OCE%%OrigDirectory    ""
%%OCE%%Directory         ""
%%OCE%%RemoveStrip      leading 0 mm trailing 0 mm
%%OCE%%Pens              off
%%OCE%%Emulation         auto
%%OCE%%Rotate            portrait
%%OCE%%ASCII             line trunc
%%OCE%%+                eol crlf
%%OCE%%+                top 10 mm
%%OCE%%+                bottom 10 mm
%%OCE%%+                left 10 mm
%%OCE%%+                right 10 mm
%%OCE%%+                fontsize 10 pt
%%OCE%%+                orientation portrait
%%OCE%%CalComp           origin lr
%%OCE%%+                checksum on
%%OCE%%+                doublesync off
%%OCE%%+                stepsize 400
%%OCE%%+                eom 3
%%OCE%%+                sync 2
```

```

%%OCE%%HPGL          plottertype HP-7586B
%%OCE%%+             ori   gin center
%%OCE%%+             sp_eof on
%%OCE%%HPGL/2        plottertype DesignJet
%%OCE%%+             origin lr
%%OCE%%+             sp_eof on

%%OCE%%Comment        Output Defaults
%%OCE%%Stamp          off
%%OCE%%Shift          0 0 mm
%%OCE%%Zoom           100 100
%%OCE%%MediaType      paper
%%OCE%%OutputSize     A0
%%OCE%%AddStrip        leading 0 mm trailing 0 mm
%%OCE%%CutMethod      synchro
%%OCE%%MediaFeed       exact
%%OCE%%Punch           off
%%OCE%%Reinforce      off
%%OCE%%Fold            off

%%OCE%%BeginBlock     1
%%OCE%%OrigDirectory  "c:\\tmp\\plots"
%%OCE%%OrigName        "scr23f.txt"
%%OCE%%Directory       "c:\\tmp\\plots"
%%OCE%%Name            "scr23f.txt"
%%OCE%%Emulation       ASCII
%%OCE%%EndBlock

%%OCE%%BeginBlock     2
%%OCE%%OrigDirectory  "c:\\tmp\\plots"
%%OCE%%OrigName        "aircraft.plt"
%%OCE%%Directory       "c:\\tmp\\plots"
%%OCE%%Name            "aircraft.plt"
%%OCE%%EndBlock

%%OCE%%BeginBlock     3
%%OCE%%OrigDirectory  "c:\\tmp\\plots"
%%OCE%%OrigName        "figb45.gpk"
%%OCE%%Directory       "c:\\tmp\\plots"
%%OCE%%Name            "figb45.gpk"
%%OCE%%EndBlock

%%OCE%%BeginBlock     4
%%OCE%%OrigDirectory  "c:\\tmp\\plots"
%%OCE%%OrigName        "draw1.clp"
%%OCE%%Directory       "c:\\tmp\\plots"
%%OCE%%Name            "draw1.clp"
%%OCE%%EndBlock

```

```

%%OCE%%BeginBlock
%%OCE%%OrigDirectory      5
%%OCE%%OrigName          "c:\\tmp\\\\plots"
%%OCE%%Directory          "c:\\tmp\\\\plots"
%%OCE%%Name               "ill03.hpg"
%%OCE%%Pens
%%OCE%%+
%%OCE%%+
%%OCE%%+
%%OCE%%EndBlock

%%OCE%%BeginOutput
%%OCE%%OutputSize         auto
%%OCE%%Fold               on
%%OCE%%IncludeBlock       1
%%OCE%%EndOutput

%%OCE%%BeginOutput
%%OCE%%Stamp              on
%%OCE%%OutputSize         auto
%%OCE%%Fold               on
%%OCE%%IncludeBlock       2
%%OCE%%EndOutput

%%OCE%%BeginOutput
%%OCE%%Stamp              on
%%OCE%%OutputSize         auto
%%OCE%%Fold               on
%%OCE%%IncludeBlock       3
%%OCE%%EndOutput

%%OCE%%BeginOutput
%%OCE%%Stamp              on
%%OCE%%OutputSize         auto
%%OCE%%Fold               on
%%OCE%%IncludeBlock       4
%%OCE%%EndOutput

%%OCE%%BeginOutput
%%OCE%%Stamp              on
%%OCE%%Zoom               auto
%%OCE%%Fold               on
%%OCE%%IncludeBlock       5
%%OCE%%EndOutput

%%OCE%%EndTicket

```

## Example 2 Matrix job

```
%%OCE%%BeginTicket 1.1
%%OCE%%Copyright 1995 Oce-Nederland B.V.
%%OCE%%CreationAppl "Oce 9800 Job Director 2.0"

%%OCE%%Comment Job Settings
%%OCE%%JobName "Frog"
%%OCE%%UserName "Dizzy Gillespie"
%%OCE%%Account "Amphibinas"
%%OCE%%Notes "Print this job before 4:02 PM\r\n"
%%OCE%%Distribution ""
%%OCE%%JobFlagSheet size A4
%%OCE%%Matrix

%%OCE%%Comment Input Defaults
%%OCE%%OrigDirectory ""
%%OCE%%Directory ""
%%OCE%%RemoveStrip leading 0 mm trailing 0 mm
%%OCE%%Pens off
%%OCE%%Emulation auto
%%OCE%%Rotate portrait
%%OCE%%Mirror off
%%OCE%%ASCII line trunc
%%OCE%%+ eol crlf
%%OCE%%+ top 10 mm
%%OCE%%+ bottom 10 mm
%%OCE%%+ left 10 mm
%%OCE%%+ right 10 mm
%%OCE%%+ fontsize 10 pt
%%OCE%%+ orientation portrait
%%OCE%%CalComp origin ul
%%OCE%%+ checksum on
%%OCE%%+ doublesync off
%%OCE%%+ stepsize 400
%%OCE%%+ eom 3
%%OCE%%+ sync 2
%%OCE%%HPGL plottertype HP-7586B
%%OCE%%+ origin center
%%OCE%%+ sp_eof on
```

```

%%OCE%%HPGL/2           plottertype DesignJet
%%OCE%%+
%%OCE%%+
%%OCE%%Comment      Output Defaults
%%OCE%%Copies        1
%%OCE%%Stamp         off
%%OCE%%Shift         off
%%OCE%%Zoom          100 100
%%OCE%%MediaType     paper
%%OCE%%OutputSize    auto
%%OCE%%AddStrip      leading 0 mm trailing 0 mm
%%OCE%%CutMethod     synchro
%%OCE%%MediaFeed     exact
%%OCE%%CopyMatrix    A0:A0 A1:A1 A2:A2 A3:A3 A4:A4
%%OCE%%Punch          off
%%OCE%%Reinforce     off
%%OCE%%Fold           off
%%OCE%%Collate        on

%%OCE%%BeginBlock 1
%%OCE%%OrigDirectory  "c:\\tmp\\plots"
%%OCE%%OrigName       "flagbody.asc"
%%OCE%%Directory      "c:\\tmp\\plots"
%%OCE%%Name           "flagbody.asc"
%%OCE%%Emulation      ASCII
%%OCE%%EndBlock

%%OCE%%BeginBlock 2
%%OCE%%OrigDirectory  "c:\\tmp\\plots"
%%OCE%%OrigName       "aircraft.plt"
%%OCE%%Directory      "c:\\tmp\\plots"
%%OCE%%Name           "aircraft.plt"
%%OCE%%EndBlock

%%OCE%%BeginBlock 3
%%OCE%%OrigDirectory  "c:\\tmp\\plots"
%%OCE%%OrigName       "figb45.gpk"
%%OCE%%Directory      "c:\\tmp\\plots"
%%OCE%%Name           "figb45.gpk"

```

```

%%OCE%%EndBlock

%%OCE%%BeginBlock 4
    %%OCE%%OrigDirectory    "c:\\tmp\\plots"
    %%OCE%%OrigName        "draw1.clp"
    %%OCE%%Directory       "c:\\tmp\\plots"
    %%OCE%%Name            "draw1.clp"
%%OCE%%EndBlock

%%OCE%%BeginBlock 5
    %%OCE%%OrigDirectory    "c:\\tmp\\plots"
    %%OCE%%OrigName        "ill03.hpg"
    %%OCE%%Directory       "c:\\tmp\\plots"
    %%OCE%%Name            "ill03.hpg"
    %%OCE%%Pens            number 1 width 0.2 mm pattern 16
    %%OCE%%+               number 2-4 width 0.3 mm
    %%OCE%%+               pattern 16
    %%OCE%%+               number 5 width 0.4 mm pattern 16
    %%OCE%%+               number 6-999 width 0.2 mm
    %%OCE%%+               pattern 14
%%OCE%%EndBlock

%%OCE%%BeginOutput 0
    %%OCE%%Copies          3
    %%OCE%%Stamp           on
    %%OCE%%Fold            on
    %%OCE%%Distribution    "To Mr Jacobs"
    %%OCE%%IncludeBlock    2 3 4
%%OCE%%EndOutput

%%OCE%%BeginOutput 1
    %%OCE%%Stamp           on
    %%OCE%%Fold            method standard
    %%OCE%%+               size 210 297 mm
    %%OCE%%+               orientation auto
    %%OCE%%+               edge 0 mm
    %%OCE%%Distribution    "To Ms Kaye"
    %%OCE%%IncludeBlock    1 5
%%OCE%%EndOutput

%%OCE%%BeginOutput 2

```

```
%%OCE%%Copies          2
%%OCE%%Distribution    "To Central Archive"
%%OCE%%Collate         off
%%OCE%%IncludeBlock    1 2 3 4 5
%%OCE%%EndOutput
%%OCE%%EndTicket
```



Océ 9800 RCF/Job Ticket

*Programmer's Manual*

---

---

---

---

## Appendix A

### List of error codes



## Set memory errors

For the Océ 9800 engines equipped with a set memory the digital sorting option is available. Digital sorting allows users to use a (scanned or sent) original repeatedly for different output options.

**Digital sorting** Digital sorting in its simplest form is setwise printing. This is possible for paper originals and for digital originals. The ‘input’ set of originals is scanned or sent once, stored in set memory and printed in the desired number of copies. Refer to 41 on page 142 for an example.

Matrix printing is advanced digital sorting. It has been described and illustrated in ‘Matrix jobs’ on page 30. For matrix jobs, the set memory is used to store digital originals.

**Set memory** Set memory size is optional 36-144 MB. The minimum amount of memory is 36 Mbyte. It can be expanded up to 144 Mbyte in steps of 36 Mbyte.

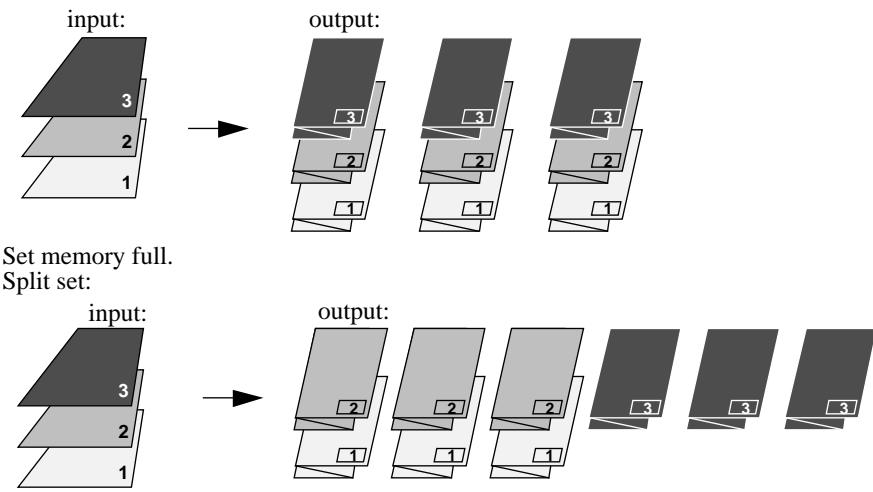
36 Mbyte can contain approximately 10 compressed E-size drawings, where this number may decrease if more information is stored in the file such as photos. When scanning or plotting jobs a part of the set memory is taken, the following errors may occur:

**job does not fit** In an empty set memory, the job you submitted does not fit. This original cannot be printed, because it cannot be stored. Advice: buy more set memory. If 144 Mbyte is available, the file is simply too large. The error message '20815 NOT\_ENOUGH\_SET\_MEMORY' will be send to the host.

## [41] Setwise printing

**Next original file of the current set does not fit** You are sending a set of files and find out that the complete set cannot be put into set memory or a plot or set does not fit in the controller's memory. The set has been put partially into memory ('20814 SET\_SPLITTED').

The controller splits the set. Refer to illustration 42. The user is informed about this by this warning message.



[42] Splitting a set, when set memory is full.

## Océ 9800 controller error codes

If an error occurs, the Océ 9800 displays an error code on the display of the operating panel. More information about the warning is displayed in the "machine menu".

For a warning there are 3 different types of recovery:

■ **Temporary Warnings (TW)**

Warnings related to a specific print file. They will not affect the functioning of the machine. The output will probably not be what the operator expected (e.g. it may be clipped, or a set may be split in two or more sets).: the operator can reset the warning message in the "machine menu".

■ **Operator Recoverable Warnings (ORW)**

Most of these warnings are communication problems between host and controller. The system administrator may have to change settings (e.g. rs232, etc.). Next, the system administrator can reset the controller-error by pushing the reset-controller-error softkey in the controller-error info screen; one controller error will be cleared at a time. Activating this function will activate all disabled host<->controller communications again, which where deactivated when a communication error was detected.

■ **Machine Recoverable Warnings (MRW)**

To recover from these warnings the operator has to switch the machine off/on. All jobs in the Océ 9800 system will be lost.

<b>error number</b>	<i>error type</i>	<i>error name</i>
20801-	MRW	Switch machine off/on
20812		

**Memory allocation warnings:**

20813	TW	Not enough controller memory; The plot file cannot be processed due to insufficient memory in the controller.
-------	----	---

---

**Memory allocation warnings:**

20814	TW	Set split; The set job does not fit completely in set or controller memory. The set will be split into two or more sub sets.
20815	TW	The plot does not fit into set memory and can therefore not be processed. Increase the amount of memory, if possible.

---

**RS-232 errors:**

20820	ORW	Framing error; Check serial communication link and settings
20821	ORW	Parity error; Check serial communication link and settings
20822	ORW	Overrun error; Check serial communication link and settings

---

**SCSI errors:**

20830	ORW	CMD group error; Check SCSI communication link and settings
20831	ORW	Check SCSI communication link and settings. The console displays 'Reset error'. Reboot the system.
20832	ORW	SCSI parity error; Check SCSI communication link and settings

---

**Data format errors:**

20860	TW	Data format not supported; Check plot file.
20861	TW	Syntax Error in data format; Check plot file.
20862	TW	Language not recognized; The ALS function was not able to detect the type of the plot file. Select the appropriate language in the RCF header.
20863	TW	The plot does not fit entirely on the selected paper size. Check RCF header and plot file.
20864	TW	Required type of color processing or required decompression method is not supported by the printer.

---

**RCF errors :**

20870	TW	Job description error; Check RCF header.
-------	----	--



---

Océ 9800 RCF/Job Ticket

---

*Programmer's Manual*

---

---

---

---

## Appendix B

### Data formats



---

## HP-GL commands

The following HP-GL commands are implemented on your Océ 9800 printer.

**Note:** *The reference for the HP-GL emulation is the HP 7586B with some extensions of the HP 7596 (e.g. real numbers for all parameters and the DV instruction).*

**Note:** *For more information, see HP 7580B, 7585B, 7586B Drafting Plotters, Interfacing and Programming Manual - Hewlett Packard. Part Number 07580-90034*

**Note:** *Commands marked with (No op) are ignored (not emulated). These instructions do not result in an error message.*

HP-GL commands	
AA Arc Absolute	AF Advance Full Page (End plot)
AH Advance Half Page (End plot)	AP Automatic pen operation (No op)
AS Pen acceleration select (No op)	AR Arc Relative
BF Buffer Plot (No op)	BL Buffered Label String
CA Designate Alternate Character Set	CC Chord angle (No op)
CI Circle	CM Character Select Mode
CP Character Plot	CS Designate Standard Character Set
CT Chord Tolerance	CV Curve generator (No op)
DC Digitize clear (No op)	DF Set to Defaults
DI Absolute Direction (Characters)	DL Downloadable Characters
DP Digitize Point (No op)	DR Relative Direction (Characters)
DS Designate Character Set	DT Define Label Terminator
DV Define Vertical Text Path	EA Edge Rectangle Absolute
EC Enable Cutter (No op)	EP Edge Polygon
ER Edge Rectangle Relative	ES Extra Space (Characters)
EW Edge Wedge	FP Fill Polygon
FR Advance Frame	FS Force select (No op)

FT Fill Type	GC Group count number (No op)
GM Graphics memory (No op)	GP Designate group (No op)
IC Input character (No op)	IM Input Masks
IN Initialize (Start plot)	IP Input P1 and P2
IV Invoke Character Set	IW Input Window
KY Assign key (No op)	LB Label ASCII String
LO Label Origin	LT Line Type
NR Not Ready	OA Output Actual Position
OB Output box (No op)	OC Output Pen Position and Status
OD Output Digitize Point (No op)	OE Output Error
OF Output Factor	OG Output count number (No op)
OH Output Hard-clip Limits	OI Output Identification
OK Output Function Key (No op)	OL Output Label Length
OO Output Options	OP Output P1 and P2
OS Output Status	OT Output Carousel Type (No op)
OW Output Window	PA Plot Absolute
PB Print Buffered Label	PD Pen Down
PG Advance Full Page (End plot)	PM Polygon Mode
PR Plot Relative	PS Plot size
PT Pen Thickness	PU Pen Up
RA Fill Rectangle Absolute	RO Rotate
RP Replot (End plot)	RR Fill Rectangle Relative
SA Select Alternate Character Set	SC Scaling
SG Select Pen Group (No op)	SI Absolute Character Size
SL Character Slant	SM Symbol Mode
SP Pen Select	SR Relative Character Size
SS Select Standard Font Set	TL Tick Length
UC User-Defined Character	UF User-Defined Fill Type
VA Activate adaptive velocity (No op)	VN Velocity normal (No op)
VS Velocity select (No op)	WD Write display (No op)

WG Fill Wedge	XT X-Tick
YT Y-Tick	

The following fonts are supported:

<i>fixed vector font</i>	<i>variable arc font</i>	<i>fixed arc font</i>	<i>name</i>
0	10	20	ANSI-ASCII
33	43	53	German
34	44	54	ISO French v1
60	70	80	ISO French v2
7	17	27	Roman Extensions
5	1	5	Special Symbols
	19	29	ISO IRV
36	46	56	ISO Italian
32	42	52	ISO Norwegian V1
39	49	59	ISO Norwegian V2
38	48	58	ISO Portuguese
30	40	50	ISO Swedish
31	41	51	ISO Swedish for names
37	47	57	ISO Spanish
35	45	55	ISO UK
110	-	-	Leroy
111	-	-	DIN 17

The default font is ANSI-ASCII (fixed vector font 0).

## HP-GL escape sequences

**Note:** Escape sequences marked with an asterisk are partially implemented.

**Note:** Escape sequences marked with (No op) are ignored (not emulated). These escape sequences do **not** result in an error message.

HP-GL escape codes	
<ESC>.A	Output Identification
<ESC>.E	Output Extended Error
<ESC>.I	Set Handshake Mode 2 *
<ESC>.K	Abort Graphic Instruction
<ESC>.M	Set Output Mode *
<ESC>.O	Output Extended Status
<ESC>.Q	Set monitor mode (No op)
<ESC>.S	Output Memory Size
<ESC>.U	End flush mode (No op)
<ESC>.Z	Programmed off Mode (No op)
<ESC>.)	Programmed off Mode (No op)
<ESC>.B	Output Buffer Space
<ESC>.H	Set Handshake Mode 1 *
<ESC>J	Abort Device Control *
<ESC>.L	Output buffer size when empty
<ESC>.N	Set Extended Output and Handshake Mode *
<ESC>.P	Set Handshake Mode *
<ESC>.R	Reset
<ESC>.T	Configure Memory *
<ESC>.Y	Programmed on Mode (No op)
<ESC>.(	Programmed on Mode (No op)
<ESC>.@	Set Plotter Configuration *

## HP-GL/2 commands

The following HP-GL/2 commands are implemented on the Océ 9800 printer.

**Note:** *The reference for HP-GL/2 is the DesignJet 650C plotter.*

**Note:** *For more information see HP-GL/2 and HP RTL Reference Guide - ISBN 0-201-63325-6.*

**Note:** *Commands marked with (No op) are ignored (not emulated). These instructions do **not** result in an error message.*

HP-GL/2 commands	
AA	Arc Absolute
AD	Alternate Font Definition
AT	Absolute Arc Three Point
CF	Character fill mode
CP	Character Plot
CT	Chord Tolerance
DF	Set to Defaults
DL	Downloadable Characters
DR	Relative Direction (Characters)
DV	Define variable text path
EC	Enable Cutter (No op)
ER	Edge Rectangle Relative
EW	Edge Wedge
FN	Select secondary font by ID (No op)
FR	Advance Frame
IN	Initialize (Start plot)
IR	Input Relative P1 and P2
LA	Line attributes (No op)
LO	Label Origin
AC	Anchor corner (No op)
AR	Arc Relative
BP	Begin Plot
CI	Circle
CR	Character range
DC	Digitize clear (No op)
DI	Absolute Direction (Characters)
DP	Digitize point (No op)
DT	Define label terminator
EA	Edge Rectangle Absolute
EP	Edge Polygon
ES	Extra Space (Characters)
FI	Select primary font by ID (No op)
FP	Fill Polygon
FT	Fill Type
IP	Input P1 and P2
IW	Input Window
LB	Label ASCII String
LT	Line Type

MC	Merge Control (No op)	MG	Message
MT	Media type (No op)	NP	Number of pens
NR	Not Ready	OD	Output digitize point (No op)
OE	Output Error	OH	Output Hard-clip Limits
OI	Output Identification	OP	Output P1 and P2
OS	Output Status	PA	Plot Absolute
PC	Pen color	PD	Pen Down
PE	Polyline Encoded	PG	Advance Full Page (End plot)
PM	Polygon Mode	PR	Plot Relative
PS	Plot size	PU	Pen Up
PW	Pen Width	QL	Quality level
RA	Fill Rectangle Absolute	RF	Raster Fill Definition
RO	Rotate	RP	Replot (End plot)
RR	Fill Rectangle Relative	RT	Relative arc three point
SA	Select Alternate Character Set	SB	Scalable or bit map fonts (No op)
SC	Scaling	SD	Standard Font Definition
SI	Absolute Character Size	SL	Character Slant
SM	Symbol Mode	SP	Pen Select
SR	Relative Character Size	SS	Select Standard font set
ST	Sort (No op)	SV	Screened Vectors
TD	Transparent data	TR	Pen Transparency
UL	User-Defined Line Type	VS	Velocity select (No op)
WG	Fill Wedge	WU	Pen Width Unit Selection

---

## HP-RTL commands

The following commands are implemented or recognized by the Océ 9800 printer.

**Note:** For more information see the HP-RTL Reference Guide -Hewlett Packard- and the HP-GL/2 and HP-RTL Reference Guide - ISBN 0-201-63325-6

HP-RTL commands	
ESC E	Reset
ESC * v # a	Set red
ESC * v # c	Set blue
ESC * v #W[data]	Configure image (No op)
ESC * b # 1	Raster line path (If the controller is in on-the-fly mode, then backward moves cannot be supported)
ESC * b #V[data]	Raster data plane (No op)
ESC * b # y	Y offset
ESC * r # C	End raster mode
ESC * r # t	Source raster height
ESC * t # r	Set graphic resolution
ESC % # B	Enter HP-GL/2 mode
ESC % # x	Universal exit
ESC * v # b	Set green
ESC * v # i	Assign color
ESC & a # h	Move current active position (CAP) horizontal (Decipoints)
ESC & a # n	Negative motion
ESC * b # m	Set compression method
ESC * b # W[data]	Transfer raster data by row/block
ESC * r # a	Start raster mode
ESC * r # s	Source raster width

ESC * t # h	Destination raster width
ESC * t # v	Destination raster height
ESC * p # p	Push/Pop Palette (No op)
ESC * p # x	Move Current Active Position (CAP) Horizontal (device units)
ESC * p # y	Move Current Active Position (CAP) Vertical (device units)
ESC * v # n	Source Transparency Mode (No op)
ESC * t # j	Render Algorithm (No op)
ESC * r # U	Simple Color Mode (No op)
ESC * v # S	Set Foreground Color (No op)

Some HP-RTL commands must end with an upper case letter, for example:  
ESC % # A.

Some HP-RTL commands may end with a lower-case or an upper-case letter,  
for example: ESC % # x or ESC % # X.

*Compressed formats:* adaptive encoding (block based) is not supported.

*Data encoding modes:* modes 2 and 3 are not supported.

*The color model:*

- Only the RGB color model is supported.
- The HP-GL/2 instructions Merge Control (MC) and Transparency (TR) define how/whether the color elements are mixed/overlapped. This is not yet supported.
- Neither the HP-GL/2 pen palette is transferred to the HP-RTL color index table, nor the reverse.
- The influence of the HP-GL/2 Quality Level instruction (QL) is ignored when setting the graphics resolution.

---

## CalComp commands

The commands interpreted by the Océ 9800 are those of Calcomp 906, Calcomp 907, Calcomp 951 and EPP (or RVI [Random Vector Interface], which is the internal controller of the Calcomp EPP Color Electrostatic Plotter). This section contains an overview of all Calcomp commands, including to which version of Calcomp they belong and how they are supported by the Océ 9800.

The information in this section applies to Poker Library 8.1 (in the controller configuration section of the print settings output of the Océ 9800 the version of the Poker Library is specified).

**Note:** *Further additional information about Calcomp can be found in the reference: 'The Calcomp Online Reference Manual, W0001-300, 1987'.*

---

### Calcomp command overview

In the table below all Calcomp commands are listed, including particularities, to which Calcomp language they belong and how they are supported on the Océ 9800. This can be:

- **normal:** the command is normally decoded and interpreted.
- **partial:** the command is normally decoded, but the interpretation is limited
- **NoOp:** the command is decoded (and also its parameters), but no interpretation is done. The command is skipped.
- **UnDec:** the command is not decoded.

Command name	version	particularies	support	code
NoOperation	906		Normal	00
SearchAddress	906	999/9999 means: EndOfPlot, 1 means: StartOfPlot	Partial	01
PenDown	906		Normal	02

PenUp	906		Normal	03
PenSelect	906	Generates a pen up, maximum number of pens is 999.	Normal	04
SymbolStringCount	906	Note the limit of the supported font (112 characters).	Normal	05
ControllerSymbol-Scaling	906		Normal	06
Radix	906		Normal	07
EscapeDisableDouble-Buffer	906	Generates an EndOfPlot!	Partial	08 00
EscapeEnableDouble-Buffer	906		NoOp	08 01
EscapeSet<cc protocol>	906		NoOp	08 {02 03 07}
EscapeSet<cc protocol>	906		NoOp	08 {04 05 06 08}
EscapeDisableDel	906		NoOp	08 09
EscapeBufferSize	906		NoOp	08 0A..08 0D
Scaling	906		Normal	09
Pause	906		NoOp	0A
PassThru_Manual	907		NoOp	0B 00 03
PassThru_NoOp	906		Normal	0B 00 04
PassThru_Plotter-Performance	906		NoOp	0B 00 05
PassThru_Newplot	906	Generates an EndOfPlot!	Normal	0B 00 06
PassThru_ChordalTolerance	907		Normal	0B 00 08 0B ...

PassThru_FontSelect	907	Only some symbols from various sets are coded. Only 0B 00 09 0B 07 0E 08 supported.	Partial	0B 00 09 0B ...
PassThru_Symb-CharacteristicsNoSlant	906		Normal	0B 05 0E
PassThru_Symb-CharacteristicsSlant	906		Normal	0B 05 0F
		plotter dependent	UnDec	0B 06
PassThru_Symbol-Scaling	907		Normal	0B 07
PassThru_DashBypass	907	May interfere with DashLine	Normal	0B 08
Circle	907	Dash circle only is dash on.	Normal	0C
DashLine	907		Normal	0D
ExtendCC_PlotSymbol	907	Only some symbols from various sets are coded.	Partial	0E 00 .. 0E 05
ExtendCC_Operator-Message	906		NoOp	0E 06
ExtendCC_OpMess-Pause	907		NoOp	0E 07
ExtendCC_Select-SymbolSetN	906	Only some symbols from various sets are coded.	Partial	0E 08 .. 0E 0C
ExtendCC_UserDef-SymbolSet	907	Fewer user symbols	Partial	0E 0D
ExtendCC_EraseUser-DefSymbolSet	907		Normal	0E 0E
ExtendCC_PlotAny-Symbol	907	Only some symbols from various sets are coded.	Partial	0E 0F

ExtendCC_Diagnostic	907		UnDec	0E 10
ExtendCC_Negate	951		NoOp	0E 11
ExtendCC_SetPen	951	Pen number ranges from 1 to 999.	Normal	0E 12
ExtendCC_PaperCut	951		NoOp	0E 13 00
ExtendCC_TopForm	951		NoOp	0E 13 01
ExtendCC_Async	951		NoOp	0E 14
ExtendCC_PatternFill	951		Normal	0E 15
ExtendCC_SetPat	951	Inside the 16..48 range.	Normal	0E 16
Reserved	951	Reserved for future usage.	NoOp	0E 17
ExtendCC_StartPlot	951		NoOp	0E 18
ExtendCC_ForcePlot	951		NoOp	0E 19
ExtendCC_PlotterSelect	951		NoOp	0E 1A
ExtendCC_PlotCopies	951		Normal	0E 1B
ExtendCC_PlotterPerformance	951		NoOp	0E 1C
ExtendCC_Mirror	951		Normal	0E 1D
ExtendCC_Window	907	Exclusive window is not supported.	Partial	0E 1E
ExtendCC_Rotation	907		Normal	0E 1F
ExtEPP_NewPen	907	Inside the 1..999 range.	Normal	0E 20
ExtEPP_PatternFill	EPP		Normal	0E 21
ExtEPP_SetPat	EPP	Inside the 16..239 range.	Normal	0E 22
ExtEPP_ColorModify	EPP	Inside the set of 1024 tints.	Normal	0E 23

ExtEPP_ColorSequence	EPP		NoOp	0E 24
ExtEPP_Disklo	EPP		NoOp	0E 25
ExtEPP_NewLevel	EPP		NoOp	0E 26
ExtEPP_PlotStatus	EPP		NoOp	0E 27
ExtEPP_SetPen	EPP	Inside the 1..999 range.	Normal	0E 28
ExtEPP_SetLevel	EPP		NoOp	0E 29
ExtEPP_RasterFill	EPP		NoOp	0E 2A
ExtEPP_Pixel	EPP	Decoding is limited to uncompressed data.	NoOp	0E 2B
ExtEPP_AreaFill	EPP		Normal	0E 2C
EndOfPlot	906		NoOp	0F
Delta	906		Normal	10 to 3F

## Command details

**The delta command** These commands (10 to 3F) have the particularities to store information concerning the parameters in the code itself. For example, the 3D command means a relative move, with 2 bytes for a negative X component and 1 byte for a positive Y component. The following matrix is used to determine the stored values (at the top the sign and number of the X components and at the left the sign and number of the Y components). See table on next page.

0x	-3	-2	-1	0	1	2	3
3	11	29	2D	1C	2C	28	10
2	31	15	39	20	38	14	30
1	35	3D	19	24	18	3C	34
0	1D	21	25		26	22	1E
-1	37	3F	1B	27	1A	3E	36
-2	33	17	3B	23	3A	16	32
-3	13	2B	2F	1F	2E	2A	12

**SearchAddress** This command refers to the 906 Calcomp controller and therefore is not a graphic command. Therefore only a few values are emulated:

- 1 which means Start Of Plot,
- 999 or 9999 which mean End Of Plot.

Each plot doesn't have to contain necessarily both Start Of Plot and End Of Plot commands. However, in case of two successive plots, at least one of the following conditions has to be met to print them separately:

- \* the first plot ends with an End Of Plot command, or
- \* the second plot begins with a Start of Plot command,
- \* each plot is preceded by and Océ 9800 RCF header.

**DisableDoubleBuffer** Double buffering is not supported. However, this command implies an End Of Plot and this is supported.

**PassThru\_FontSelect** This command select a set of characters. Only the Calcomp 907 Symbol set 0, characters 0 to 95, and the Calcomp 907 Downloaded Symbol set, characters 288 to 383. Further the characters 384 to 479 of this same set are mapped onto the characters 288 to 383. All not supported characters are printed as white space (blank).

Note that the Calcomp 907 Symbol set 0 contains characters from various sets of Calcomp symbols.

**ExtendCC\_plotSymbol, ExtendCCSelectSymbolSetN,**

**ExtendCC\_PlotAnySymbol** The support of these commands is limited to the supported fonts or characters.

**ExtendCC\_UserDefSymbolSet** With this command the user can define his own symbols. Instead of 192 symbols, only 96 symbols are supported. Note that all definitions are accepted, but the second set of 96 symbols overrides the first one.

**ExtEPP\_SetPat, ExtendCC\_SetPat:** Calcomp defines two sets of patterns. Number 1 through 15 are Calcomp language specific and number 240 to 255 are used to generate 16 different grey levels. Therefore, number 16 thru 239 are left open to the user.

The ExtendCC\_SetPat command can be used for the patterns 16 thru 48, where the ExtEPP\_SetPat command can be used for the patterns 16 thru 239.

Addressing these patterns depends on which command is used: if you want to modify pattern #17 with the ExtEPP\_SetPat command, you have to specify number 2, because the 15 predefined patterns have to be used as offset. If you want to modify this same pattern with ExtendCC\_SetPat command, you have to use number 17!

**ExtendCC\_Window** Calcomp defines inclusive (all vectors inside the window are plotted) and exclusive windows (all vectors outside the window are plotted). The exclusive option is not supported by the Océ 9800.

**Note:** *Sending a window command with identical pairs of coordinates disables windowing. This feature is **not** explained in the Calcomp reference documentation!*

**ExtEPP\_SetPen, ExtendCC\_SetPen** These commands allow the user to specify a pen number ranging from 1 to 999 (although the documentation says only 1 to 16). In the pen colors some discrepancies may appear.

**PenSelect, ExtendCC\_NewPen** Both commands accept a pen number parameter ranging from 1 to 999, although the Calcomp documentation specifies only the range 1 to 16 for the PenSelect command. If the ExtendCC\_NewPen number parameter ranges from 1000 to 1024, pen #1 is selected. Selecting pen #0 by either command is ignored.

At initialisation time a set of pens and patterns are created. On top of the 15 patterns defined by Calcomp, a set of 16 bitmaps sized 32 by 32 pixels are created to reproduce 16 grey shades.

**Note:** *The toggle between the default pen palette and the software pen palette is not supported by the Océ 9800.*

**EndOfPlot** This command (code 0F) is decoded, but generates a NoOperation. The remaining characters (up to the end of the Calcomp message) are skipped.

**Radix** The radix is initialized with the value 64, according to the Calcomp requirements. It is saved between two plots until another Radix command is processed. The several Calcomp plotters have a different behaviour: some do the same as the Océ 9800 (saving the currently used Radix between two plots), others return to the default value after each plot.

The Calcomp documentation advises to set the radix value at the beginning of each plot.

## Error conditions

A syntax error (error code 20861 on the Océ 9800) is generated if the Command Code is not valid.

## Interface requirements

The Calcomp language uses some specific characters to identify messages. The number and the values of the synchronisation characters (one or two), the presence of the checksum in the message and the end-of-message character are user defined. Also the step value is user defined. It is used as a scaling factor and is necessary to the dash definition.

All these 5 parameters are required in order to correctly plot the Calcomp file (i.e. these values should be specified in the Océ 9800 Remote Controle File header sent before the plot, or the default values of these parameters on the Océ 9800 should be configured correctly).

---

## ASCII emulation

The Océ 9800 prints ASCII text files. ASCII codes in the range 0-127 are supported.

Hex Code	Emulation
\$00 NUL	- (not emulated)
\$01 SOH	-
\$02 STX	-
\$03 ETX	-
\$04 EOT	emulated
\$05 ENQ	-
\$06 ACK	-
\$07 BEL	-
\$08 BS	partially emulated
\$09 HT	emulated
\$0A LF	emulated
\$0B VT	-
\$0C FF	emulated
\$0D CR	emulated
\$0E SO	-
\$0F SI	-
\$10 DLE	-
\$11 DC1	-
\$12 DC2	-
\$13 DC3	-
\$14 DC4	-
\$15 NAK	-
\$16 SYN	-
\$17 ETB	-
\$18 CAN	-
\$19 EM	-
\$1A SUB	-
\$1B ESC	-
\$1C FS	-
\$1D GS	-
\$1E RS	-
\$1F US	-
\$20 SP	emulated
\$21 !	emulated
... ...	emulated (from \$22 to \$7E)
\$7F DEL	-

### Hexadecimal - Character table

00	NUL	01	SOH	02	STX	03	ETX	04	EOT	05	ENQ	06	ACK	07	BEL
08	BS	09	HT	0A	NL	0B	VT	0C	NP	0D	CR	0E	SO	0F	SI
10	DLE	11	DC1	12	DC2	13	DC3	14	DC4	15	NAK	16	SYN	17	ETB
18	CAN	19	EM	1A	SUB	1B	ESC	1C	FS	1D	GS	1E	RS	1F	US
20	SP	21	!	22	"	23	#	24	\$	25	%	26	&	27	'
28	(	29	)	2A	*	2B	+	2C	,	2D	-	2E	.	2F	/
30	0	31	1	32	2	33	3	34	4	35	5	36	6	37	7
38	8	39	9	3A	:	3B	;	3C	<	3D	=	3E	>	3F	?
40	@	41	A	42	B	43	C	44	D	45	E	46	F	47	G
48	H	49	I	4A	J	4B	K	4C	L	4D	M	4E	N	4F	O
50	P	51	Q	52	R	53	S	54	T	55	U	56	V	57	W
58	X	59	Y	5A	Z	5B	[	5C	\	5D	]	5E	^	5F	-
60	'	61	a	62	b	63	c	64	d	65	e	66	f	67	g
68	h	69	i	6A	j	6B	k	6C	l	6D	m	6E	n	6F	o
70	p	71	q	72	r	73	s	74	t	75	u	76	v	77	w
78	x	79	y	7A	z	7B	{	7C		7D	}	7E	~	7F	DEL

**Note:** An ASCII file must always end with EOT.

**Note:** ASCII is not recognized automatically. You must specify APPLDATA 056 (see 'ASCII emulation (APPLDATA 056)' on page 65).

---

## TIFF

**Note:** For more information , see the *TIFF specification, Revision 6.0, Aldus Developers Desk, June 3, 1992.*

If a section is mentioned in the remarks below, it refers to the reference document mentioned above.

- Multiple images in one file (multiple tiff) is not supported.
- Unknown fields/tags are skipped without generating an error.
- Grayscale images (section 4), palette-color images (section 5), and RGB Full color images (section 6) are ignored.
- The Additional Baseline TIFF Requirements (section 7):  
The ASCII field type, the floating point data type, the new data type, and the new pixel types are not supported.
- Baseline Field Reference Guide (section 8): The following fields are not supported (or partially if indicated):
  - Artist (ASCII type)
  - ColorMap
  - Copyright (ASCII type)
  - DataTime (ASCII type)
  - ExtraSamples
  - FillOrder: only value 1 is supported
  - FreeByteCounts
  - FreeOffsets
  - GrayResponseCurve
  - GrayResponseUnit
  - HostComputer (ASCII type)
  - ImageDescription (ASCII type)
  - Make (ASCII type)
  - MaxSampleValue
  - MinSampleValue
  - Model (ASCII type)
  - Orientation: only the default value 1 is supported.
  - PhotometricInterpretation: only the values 0 and 1 are supported
  - PlanarConfiguration
  - Software (ASCII type)
  - Thresholding
- LZW compression (section 13): this compression method is not supported (the warning 20864 will be generated by the controller).

- Differencing Predictor (section 14): This is related to the LZW compression and therefore not supported.
- CMYK Images (section 16), HalftoneHints (section 17) and Associated Alpha Handling (section 18) are not supported.
- Data Sample Format (section 19): only SampleFormat = 1 is supported. The SMinSampleValue and SMaxSampleValue fields are ignored.
- RGB Image Colorimetry (section 20), YCbCr Images (section 21), JPEG compression (section 22) and CIE L\*a\*b\* images (section 23) are not supported.

---

## CALS

**Note:** More information can be found in the following documents:

- MIL-STD-1840B, Military Standard, Automated Interchange of Technical Information, 3 November 1992.
- MIL-R-28002B, Military specification, Raster Graphic Representation in Binary Format, 14 December 1992
- CCITT Recommendation T.6: 1988, Facsimile Coding Schemes and Coding Control Functions for Group 4 Facsimile Apparatus.

- CALS raster type II is not supported. Thus only rtype=1 is supported.  
Note that dtype is equivalent to rtype.
- The "MIXED" character string is not supported in the rorient and the rpelcnt records (it is ignored).
- The two values of the rpelcnt record are interpreted as the width and length of the image (in pixels).
- If the rdensity record is not present, then 400 dpi is assumed.
- The rdensity record is equivalent to the rdensity record.
- If the controller encounters the specversion record, then the file is identified as a CALS file. The attribute of this record is ignored. This record is hardly used.
- The srcrefid record also identifies the file as a CALS file. Its attribute is ignored.
- The attribute of the dstsrcrefid, doccls, and the notes record is ignored.
- The datfilid, didid, fosipubid, moduleid, txtfilid, figid and srcgph records are ignored.

Océ 9800 RCF/Job Ticket

*Programmer's Manual*

---

---

---

---

## Appendix C

## RCF grammar

*Backus-Naur Form (BNF) description of RCF.*



---

## RCF grammar

This paragraph describes the syntax of the Remote Control Format. It is based on the Clear Text Encoding format of the ANSI X3.122-1986 standard named "Computer Graphics Metafile for the Storage and Transfer of Picture Description Information". Note that:

- the RCF grammar does not contain all the ins and outs of the ANSI format, only a subset.
- the RCF uses the ASCII encoding standard.
- everything in an RCF that is not inside a string or a comment is case-insensitive.

**Note:** See chapter 2, 'Remote Control Format (RCF)' on page 15 for details about the supported types and limitations.

**Conventions** The grammar is given in Backus-Naur Form (BNF):

<symbol>	non-terminal
<symbol>*	star closure (0 or more occurrences)
<symbol>+	plus closure (1 or more occurrences)
...   ...	... or ...
... ::= ...	... is defined as ...
/* ... */	comment, not part of the grammar.
ascii n	character indicated by the number n in the ASCII encoding.
"string"	literal string that occurs as is in the language.

**Grammar** The RCF header is defined as follows:

```
<rcf_header>      ::= <begin_metafile>
                      <metafile_identifier>
                      <metafile_descriptor>
                      <metafile_contents>
                      <end_metafile>
<begin_metafile>  ::= "BEGMF"
                      <optsep>
                      <string> /* user defined comment */
                      <term>
```

```

<end_metafile>      ::= "ENDMF" <term>
<metafile_identifier> ::= "MFVERSION"
                        <softsep>
                        <version>
                        <term>
<metafile_descriptor> ::= "MFDESC"
                        <optsep>
                        <mfdesc_string> /* description */
                        <term>
<metafile_contents> ::= <application_data>*
<application_data>  ::= "APPLDATA"
                        <softsep>
                        <num> /* identifier */
                        <optsep> | <hardsep> >
                        <appl_string>
                        <term>
<appl_string>       ::= <single_quote> <datarecord> <single_quote>
                        | <double_quote> <datarecord> <double_quote>
<term>              ::= <optsep>
                        <slash> | <:semicolon>
                        <optsep>
<sepchar>           ::= <space>
                        | <line_feed>
                        | <horizontal_tab>
                        | <carriage_return>
<softsep>           ::= <sepchar>+
<optsep>           ::= <sepchar>*
<hardsep>           ::= <optsep> <comma> <optsep>
<digit>             ::= "0" | "1" | "2" | "3" | "4" | "5" | "6" | "7" | "8" | "9"
<letter>             ::= "a" | "b" | ... | "Y" | "z" | "A" | "B" | ... | "Y" | "Z"
<alphanumeric>      ::= <digit> | <letter>
<num>               ::= <digit>+
<space>             ::= " "          /* ascii 32 */
<double_quote>       ::= """          /* ascii 34 */
<percent_sign>      ::= "%"          /* ascii 37 */
<single_quote>       ::= "'"          /* ascii 39 */
<slash>              ::= "/"          /* ascii 47 */
<semicolon>          ::= ";"          /* ascii 59 */
<horizontal_tab>    ::= ascii 9
<line_feed>          ::= ascii 10
<vertical_tab>      ::= ascii 11
<form_feed>          ::= ascii 12
<carriage_return>   ::= ascii 13

```

**String types** String parameters are represented as character strings immediately surrounded by a pair of either single- or double-quote characters. The string itself cannot contain a single-quote when is surrounded by single-quote characters, the same holds for the use of a double quote in a string surrounded by double-quote characters.

```

<string>      ::= <single_quote> <str_no_single> <single_quote>
                  | <double_quote> <str_no_double> <double_quote>
<str_no_single> ::= < <string_char> | <double_quote> >*
<str_no_double> ::= < <string_char> | <single_quote> >*
<string_char>  ::= <alphanumeric> | <sepchar> | <equal> | <comma> | <dash> =>
                  | <less_than> | <greater_than>

```

**Metafile descriptor string** This string is used to recognise the Oce RCF headers:

```

<mfdesc_string> ::= <single_quote> <oce_ident> <single_quote>
                  | <double_quote> <oce_ident> <double_quote>

```

For RCF version 1 headers, the <oce\_ident> string is

```

<oce_ident>      ::= "Oce Graphics Configuration format V1.00"

```

For RCF version 2 and higher, this becomes

```

<oce_ident>      ::= "Oce RCF" <hardsep> <version> <hardsep> <type> <hardsep>=>1
                  <unit>
<version>        ::= "Version=" <fixed_point>
<type>           ::= "Type=Header"
                  | "Type=MatrixPrg"
                  | "Type=ScanToFile"
<unit>           ::= "Unit=Metric"
                  | "Unit=Inches"
<fixed_point>    ::= <digit>+
                  | <digit>+ <point> <digit>*

```

**Note:** The whole <oce\_ident> string is case-insensitive, including the type and unit settings.

1. A “=>” is not part of the grammar, it means that the text continues on the next line

**DataRecord** The contents of the datarecord which occurs as a string in the grammar above has the following syntax:

```
<datarecord>      ::= <optsep> /* may be empty */  
                  | <optsep> <param_list> <optsep>  
<param_list>    ::= <param>  
                  | <param> <hardsep> <param_list>  
<param>         ::= <key> <equal> <value>  
<key>           ::= <letter> <alphanumeric>  
<value>         ::= <num>  
                  | <num> <dash> <num>  
                  | <fixed_point>  
                  | <list>  
<list>          ::= <less_than> <optsep> <numlist> <optsep> <greater_than>  
<numlist>       ::= <num>  
                  | <num> <listsep> <numlist>  
<listsep>       ::= <softsep> | <hardsep>  
<comma>         ::= ","          /* ascii 44 */  
<dash>          ::= "-"          /* ascii 45 */  
<less_than>     ::= "<"          /* ascii 60 */  
<equal>          ::= "="          /* ascii 61 */  
<greater_than>  ::= ">"          /* ascii 62 */
```

**Note:** *The whole data-record is case-insensitive.*

**Comments** Comments are encoded as a series of printing characters and <sepchar>'s surrounded by "%" (percent-sign) characters. The text of the comment may not include this comment delimiter character.

Comments may be included in any place that a separator (indicated by <softsep>, so also as part of a <hardsep>), may be used, and are equivalent to a <softsep>; they may be replaced by a space-character in parsing, without affecting the meaning of the metafile.



---

Océ 9800 RCF/Job Ticket

---

*Programmer's Manual*

---

---

---

---

## Appendix D

## Miscellaneous



---

## Notation conventions

There are a number of notation conventions used in this manual. This consistent style enables you to quickly become conversant with the use of this manual and consequently the Océ 9800 RCF/Job Ticket.

**Description** Each section or subsection contains a description of the feature or operation identified in the title. It might also include possible applications, as well as any guidelines that you should bear in mind.

**Figures and tables** Figures and tables are titled and numbered sequentially throughout this manual. Figures include pictures of product components, examples, and diagrams of concepts discussed in the description.

**Attention getters** There are several types of information to which we draw your attention. This information is classified as follows:

**Note:** *In a 'Note', information is given about matters which ensure the proper functioning of the copier, but useful advice concerning its operation may also be given.*

---

---

## Reader's comment sheet

Have you found this manual to be accurate?

- Yes
- No

Could you operate the product after reading this manual?

- Yes
- No

Does this manual provide enough background information?

- Yes
- No

Is the format of this manual convenient in size, readability and arrangement  
(page layout, chapter order, etc.)?

- Yes
- No

Could you find the information you were looking for?

- Always
- Most of the times
- Sometimes
- Not at all

What did you use to find the required information?

- Table of contents
- Index

Are you satisfied with this manual?

- Yes
- No

Thank you for evaluating this manual.

If you have other comments or concerns, please explain or suggest  
improvements overleaf or on a separate sheet.

2998520

### Comments:

Date:

This reader's comment sheet is completed by:  
(If you prefer to remain unknown, please do fill in your occupation)

Name: \_\_\_\_\_

### Occupation:

### Company:

Phone:

### Address:

City

## Country

Please return this sheet to:

Océ-Nederland B.V.  
For the attention of ITC user documentation.  
P.O. Box 101,  
5900 MA Venlo  
The Netherlands

---

## Related documentation

- Océ 9800 Copier/Printer User Manual
- Océ 9800 Printer User Manual
- Océ 9800 Repro Station User Manual
- Océ 9800 Job Director User Manual
- HP-GL
  - HP 7475A Interfacing and Programming Manual, Hewlett Packard, Part nr. 07475-90001, HP 7585/86 Interfacing and Programming Manual, Hewlett Packard.
- HP-GL/22
  - HP-GL/2 Reference Guide, Handbook for Program Developers, 1990, Hewlett Packard, Part nr. 5959-9733.
- HP-RTL
  - HP-RTL Reference Guide, September 1991, Hewlett Packard, Part nr. 5960-3743.
- CALS
  - MIL-STD-1840A CALS USA military standard of 22 December 1987 for data format for graphical data exchange, and related
  - MIL-R-28002A of 20 December 1988 and revised for CLASS A data
  - MIL-R-28002B military standard of 30 December 1989.
- Calcomp
  - Calcomp Plotter Interfacing and Programming Manual, California Computer Products inc., 1986, Part nr. W0001-300.
- TIFF
  - TIFF Revision 6.0 of June 1992, Aldus Corporation, Seattle.

**Note:** Not all TIFF 6.0 features are supported by the Océ 9800 printer.



# Index

## A

account id 27  
Account, 9800 Job Ticket 112  
AccountId 26  
AddStrip, 9800 Job Ticket 112  
AF Auto Format APPLDATA 021 41  
AFNOR 44  
AH Arae Height APPLDATA 025 51  
AI Account ID APPLDATA 002 26  
APPLDATA 18  
AR Auto Rotate APPLDATA 003 35  
AreaHeight 51  
AreaWidth 51  
AreaX 51  
AreaY 51  
AS Auto Scale APPLDATA 003 35  
ASCII 65  
    commands 164  
ASCII, 9800 Job Ticket 113  
AutoFormat 41, 42  
automatic language sensing (ALS) 69  
AutoRotate 35  
AutoScale 35  
AW Area Width APPLDATA 025 51  
AX AreaX APPLDATA 025 51  
AY Area Y APPLDATA 025 51

## B

BeginBlock, 9800 Job Ticket 109  
BeginOutput, 9800 Job Ticket 109  
BeginTicket, 9800 Job Ticket 109  
BEGMF 17  
belt 47  
BI Binding Edge APPLDATA 022 44  
bidsets 59  
BindingEdge 44  
BM Bottom Margin APPLDATA 056 65  
bottom 66  
BottomMargin 65

BY Bypass APPLDATA 021 41  
BYpass 41

## C

CalComp 66  
    commands 156  
CalComp, 9800 Job Ticket 113  
carriage return 66  
ChecKsum 67  
CI Cut Info APPLDATA 023 46  
CK Checksum APPLDATA 058 67  
CL Cut Length APPLDATA 023 46  
CM Copy Method APPLDATA 002 26  
CO Copies APPLDATA 002 26  
collate, 9800 Job Ticket 114  
Comment, 9800 Job Ticket 115  
controller error codes 144  
COpies 26  
Copies, 9800 Job Ticket 115  
Copymatrix, 9800 Job Ticket 115  
CopyMethod 26, 27  
Copyright, 9800 Job Ticket 117  
CreationAppl, 9800 Job Ticket 117  
CutInfo 46  
CutLength 46  
CutMethod, 9800 Job Ticket 117

## D

data formats 147  
DB Double\_sync APPLDATA 058 67  
DE Deposit APPLDATA 023 46  
delivery 46  
DEposit 46  
digital sorting 142  
DIN (European) 53  
Directory, 9800 Job Ticket 118  
Distribution, 9800 Job Ticket 118  
DouBle\_sync 67

## E

edge correction 49

EL End of Line APPLDATA 056 65  
EM End of message APPLDATA 058 67  
Emulation, 9800 Job Ticket 118  
EndBlock, 9800 Job Ticket 110  
ENDMF 17, 22  
EndofLine 65  
EndofMessage 67  
EndOutput, 9800 Job Ticket 110  
EndTicket, 9800 Job Ticket 110  
ERICSSON 44  
error  
    codes 141

**F**  
finishing 44  
fixed-point 19  
FM Framing Mode APPLDATA 025 51  
FO Fold Options APPLDATA 022 44  
fold 44  
    methods 44  
    orientation 45  
    packet length 45  
    packet width 45  
Fold, 9800 Job Ticket 119  
foldMEthod 44  
FoldOptions 44  
FoldOrientation 44  
FontSize 39, 65  
format 53  
FR Fold Orientation APPLDATA 022 44  
FramingMode 51  
FS Font Size APPLDATA 020 39  
FS Font Size APPLDATA 056 65

**G**  
GreyScale 39  
GS Grey Scale APPLDATA 020 39

**H**  
header 17  
HP-GL 61

    commands 148  
    escape sequences 151  
HPGL, 9800 Job Ticket 119  
HP-GL/2 63  
    commands 152  
HPGL/2, 9800 Job Ticket 120  
HP-RTL  
    commands 154

**I**  
IncludeBlock, 9800 Job Ticket 105, 110  
integer 18

**J**  
JB Job Boundary APPLDATA 002 26  
job  
    parameters 26  
    ticket format 89  
Job Ticket, 9800 - 89  
JobBoundary 26, 27  
JobCollate, 9800 Job Ticket 120  
JobFlagsheet, 9800 Job Ticket 120  
JobName, 9800 Job Ticket 121

**K**  
keywords overview, 9800 Job Ticket 129

**L**  
LA LeadingedgeAdd APPLDATA 024 49  
LanGuage 69  
large ANSI (American) 53  
LC Legend Control APPLDATA 003 35  
LE fold packet Length APPLDATA 022 44  
LeadingedgeAdd 49  
LeadingedgeRemove 49  
LeftMargin 65  
LeftRightalignment 58  
LegendControl 35  
LEngth 44

LG Language APPLDATA 059 69  
line feed 66  
LineOverflow 65  
list 19  
LM Left Margin APPLDATA 056 65  
LO Line Overflow APPLDATA 056 65  
LR Leadingedge remove APPLDATA 024 49  
LR Left Right alignment APPLDATA 029 58  
LUI 12

OR Orientation APPLDATA 056 65  
OR Origin APPLDATA 050 61  
OR Origin APPLDATA 051 63  
OR plot origin APPLDATA 058 67  
ORIENTATION 65  
OrigDirectory, 9800 Job Ticket 123  
ORIGIN 61, 63, 67  
Original to Copy Matrix 31  
OrigName, 9800 Job Ticket 124  
OutputSize, 9800 Job Ticket 124  
OverlayBlock, 9800 Job Ticket 106

---

## M

manual feed 42  
margins 66  
    bottom 66  
    right 66  
matrix  
    jobs 16, 30  
    program 30  
    programs 10  
MatrixPrg 17  
ME fold method APPLDATA 022 44  
ME Media APPLDATA 021 41  
MEdia 41  
media  
    selection 41, 42  
MediaFeed, 9800 Job Ticket 121  
MediaType, 9800 Job Ticket 122  
MFDESC 17, 22  
MFVERSION 17  
miscellaneous 175  
MP 10

---

## N

Name, 9800 Job Ticket 123  
notation conventions 83, 176  
Notes, 9800 Job Ticket 123  
number of copies 26

---

## O

operator  
    recoverable warnings 144

---

## P

PaperFormat 41  
pen  
    attributes 24  
PenNumber 24  
PenPattern 24  
PenPriority 63, 67  
Pens, 9800 Job Ticket 124  
PenWidth 24  
PF Paper Format APPLDATA 021 41  
PL PLot List APPLDATA 002 26  
plot  
    list 27  
    number 27  
    origin 67  
PlotList 26, 27  
PlotNumber 26, 27  
plotter 61  
plotterTYpe 61, 63  
PN Pen Number APPLDATA 001 24  
PN Plot Number APPLDATA 002 26  
PO Position APPLDATA 020 39  
POsition 39  
PP Pen Pattern APPLDATA 001 24  
PP Pen Priority APPLDATA 051 63  
PP Pen Priority APPLDATA 058 67  
punch 44  
Punch, 9800 Job Ticket 125  
PW Pen Width APPLDATA 001 24

---

**R**

range 19  
RCF  
    structure 17  
RCH 10  
Remote  
    Configuration Header 10  
remote 68  
RemoveStrip, 9800 Job Ticket 125  
Repro Station 90  
RightMargin 65  
RM Right Margin APPLDATA 056 65  
RM Rolloverruling Method APPLDATA 021 41  
RO Rotation APPLDATA 003 35  
RolloverrulingMethod 41, 42  
Rotate, 9800 Job Ticket 125  
ROtation 35

SP0\_eof 61, 63  
SR Shift Right APPLDATA 003 35  
SS Stamp Selection APPLDATA 020 39  
ST Step APPLDATA 058 67  
stamp 39  
    font size 40  
    grey scale 40  
    position 39  
Stamp, 9800 Job Ticket 126  
StampSelection 39  
standard 44  
standard cut 47  
STep 67  
steps per inch 67  
string 19  
    number 39  
StringNumber 39  
SU Shift Up APPLDATA 003 35  
SY Sync\_code APPLDATA 058 67  
SYnc\_code 67  
synchro cut 47

---

**S**

S1 format APPLDATA 026 53  
S2 format APPLDATA 026 53  
S3 format APPLDATA 026 53  
S4 format APPLDATA 026 53  
scale 35  
SD Shift Down APPLDATA 003 35  
set  
    jobs 16  
    memory 142  
    of plot file jobs 28  
shift 35  
Shift, 9800 Job Ticket 126  
ShiftDown 35  
ShiftLeft 35  
ShiftRight 35  
ShiftUp 35  
single  
    job 16, 27  
    plot file jobs 28  
SL Shift Left APPLDATA 003 35  
small ANSI (American) 53  
SN String Number APPLDATA 020 39  
SO format APPLDATA 026 53  
SP SO eof PPLDATA 050 61  
SP SP0 eof APPLDATA 051 63

TA Trailing edge Add APPLDATA 024 49  
TB Top Bottom alignment APPLDATA 029 58  
temporary  
    warnings 144  
TM Top Margin APPLDATA 056 65  
TopBottomalignment 58  
TopMargin 65  
TR Trailing edge Remove APPLDATA 024 49  
TrailingedgeAdd 49  
TrailingedgeRemove 49  
transformations 35  
truncated 66  
TY Plotter type APPLDATA 050 61  
TY plotter type APPLDATA 051 63  
type 61  
Type, 9800 Job Ticket 127

---

**U**

UI User Id APPLDATA 002 26  
UnderlayBlock, 9800 Job Ticket 106

user id 27  
UserId 26  
UserName, 9800 Job Ticket 127

---

**W**

WI Width APPLDATA 022 44  
WIdth 44  
wrapped 66

---

**X**

XS X Scale APPLDATA 003 35  
XScale 35

---

**Y**

YS Y Scale APPLDATA 003 35  
YScale 35

---

**Z**

Zoom, 9800 Job Ticket 127